



HAL
open science

Thick Fuzzy Sets (TFSs) and Their Potential Use in Uncertain Fuzzy Computations and Modeling

Reda Boukezzoula, Luc Jaulin, Benoît Desrochers, Didier Coquin

► **To cite this version:**

Reda Boukezzoula, Luc Jaulin, Benoît Desrochers, Didier Coquin. Thick Fuzzy Sets (TFSs) and Their Potential Use in Uncertain Fuzzy Computations and Modeling. *IEEE Transactions on Fuzzy Systems*, 2021, 29 (11), pp.3334 - 3348. 10.1109/tfuzz.2020.3018550 . hal-03128908

HAL Id: hal-03128908

<https://ensta-bretagne.hal.science/hal-03128908v1>

Submitted on 2 Feb 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Thick Fuzzy Sets (TFSs) and Their Potential Use in Uncertain Fuzzy Computations and Modeling

Reda Boukezzoula ^(a), Luc Jaulin ^(b), Benoit Desrochers ^(b) and Didier Coquin ^(a)

Abstract—This paper aims at proposing the concept of thick fuzzy sets (TFSs). A TFS is based on the joint use of thick sets (TSs) and α -cuts concepts. A TFS is represented by a family of nested TSs. A TS is an uncertain set, which is represented by a pair of crisp sets (CSs). These CSs characterize the upper and lower bounds of the TS. Therefore, a TS can be regarded as an interval of CSs. In this framework, as a type-1 fuzzy set (T1FS) is regarded as a family of nested CSs, a TFS can be represented by a family of nested TSs. Furthermore, according to the vertical dimension α , a TFS can be regarded as an interval with T1FS bounds. The potentialities of the TFS concept have been validated using application examples where a real-world application for modeling the zone explored by an underwater robot is given.

Index Terms—Type-1 fuzzy sets (T1FSs), Type-2 fuzzy sets (T2FSs), Crisp sets (CSs), Thick sets (TSs), α -cuts principle, Thick fuzzy sets (TFSs), underwater robot.

I. INTRODUCTION

The concept of type-2 fuzzy sets was initially proposed by Zadeh [1] as an extension of conventional fuzzy sets. This extension was intended for dealing with uncertainty. In this paper, the conventional fuzzy sets are referred to as type-1 fuzzy sets (T1FSs). The denomination type-1 initially proposed by Zadeh [1] is used here to distinguish between other fuzzy extensions, such as T2FSs. Therefore, T2FSs have been proposed to model uncertainties in T1FSs.

A T2FS is generally defined by these lower and upper T1FSs and is subjected to the inclusion constraint between them—i.e., lower T1FS is included in the upper T1FS. In the T2FS representation, the footprint of uncertainty (FOU) characterizes the blurring of a T1FS [2] and is delimited by the two bounding lower and upper T1FSs. Therefore, the uncertainty in T2FSs is exhibited through the FOU concept.

Over the past 30 years, interest in T2FSs has grown significantly thanks to the research works of Mendel et al. (e.g. [3][4][5][6]), Castillo et al. (e.g. [7][8][9]), etc and many others. Therefore, since Zadeh’s innovative ideas, research on T2FSs has expanded and it is now difficult to draw up an exhaustive list of all the works that have been published in the literature. Unfortunately, many excellent works are missed in this paper (see [10] for a retrospective). Furthermore, the

literature is unanimous about the importance of T2FSs for dealing with uncertainty when the manipulated information cannot be represented by the only T1FSs. The usefulness and potentialities of T2FSs have been demonstrated through applications in many domains: multicriteria decision making [11][12][13][14], image processing and pattern recognition [15][16], automatic control (e.g. [7][8][17][18][19][20]), etc.

In fuzzy theory and its applications, two equivalent vertical and horizontal representations are available for dealing with fuzzy sets [21]. For instance, a T1FS can be represented either by a membership function (vertical representation) or by a family (a system) of nested crisp sets (CSs), called α -levels (α -cuts) sets (horizontal representation) [22][23][24]. The membership function method (the functional approach) is, by far, the most widely used in the literature. However, in some situations, it may be more suitable to use the α -level CSs (the set-method) [25][26]. Each of these two representations has its advantages in the study of fuzzy sets. One of the main advantages of the set-based approach is that it explicitly establishes a connection between fuzzy sets and CSs and their manipulation through interval arithmetic (IA) and interval reasoning. In this paper, the set-method (α -cuts approach) is adopted. In literature, the set-method has been extended to T2FSs. Historically, Zadeh [1] was the pioneer who defined operations for T2FSs using α -cuts. Since then, several representations and computational methods based on α -cut and α -plane principles have been proposed (e.g., [27][28][29]).

To challenge the set-method for dealing with uncertain T1FSs, the concept of thick fuzzy sets (TFSs) is proposed in this paper. This concept is based on the combination of thick sets [30][31] and α -cuts principles. Therefore, a TFS is regarded as a family of nested TSs.

A CS \mathbb{X} has sharp boundaries, i.e., there are two possibilities only: an element x belongs to \mathbb{X} ($x \in \mathbb{X}$), or an element does not belong to \mathbb{X} ($x \notin \mathbb{X}$). Therefore, the belonging of an element that is inside \mathbb{X} is certain and the belonging of an element that is outside \mathbb{X} is impossible. In circumstances when uncertainty about the belonging of an element in a CS is observed (x is perhaps in \mathbb{X}), this situation cannot effectively be represented by a single CS. For dealing with this situation, a TS representation can be used [30][31]. A TS is an uncertain set that is denoted by $[\mathbb{X}]$ and defined by two CSs \mathbb{X}^{inf} and \mathbb{X}^{sup} respectively such as $\mathbb{X}^{\text{inf}} \subset \mathbb{X}^{\text{sup}}$. These CSs are regarded as lower and higher bounds of \mathbb{X} (see Fig. 1). Therefore, the TS $[\mathbb{X}]$ can be represented as an interval of CSs where \mathbb{X}^{inf} and

The authors are with : (1) the LISTIC, Université Savoie Mont Blanc, 5 Chemin de Bellevue, 74940 Annecy-le-Vieux, France and (2) the Lab-STICC, ENSTA-Bretagne, 2 rue François Verny, 29806, Brest, France. Corresponding author: reda.boukezzoula@univ-smb.fr.

\mathbb{X}^{sup} are its bounds, i.e., $[\mathbb{X}] = [\mathbb{X}^{\text{inf}}, \mathbb{X}^{\text{sup}}]$. The difference $\mathbb{X}^? = \mathbb{X}^{\text{sup}} \setminus \mathbb{X}^{\text{inf}}$ that represents the uncertainty is called the penumbra.

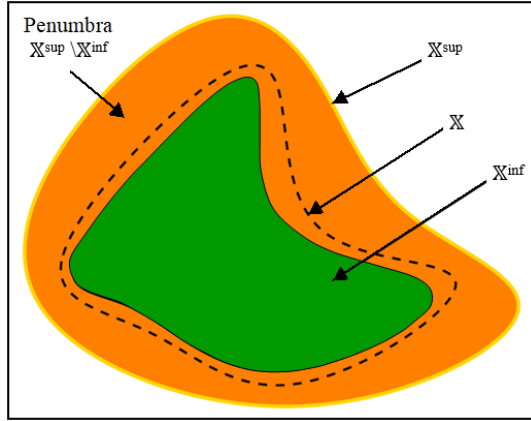


Fig. 1: The representation of the thick set (TS) $[\mathbb{X}]$

A fuzzy set (a T1FS) is a generalization of a CS. Thus, T1FSs were introduced as an extension of the conventional notion of CSs. From a philosophical point of view, the TFS formalism is based on a set-method approach according to the α -cuts principle. The TFS representation principle is announced as follows: as a T1FS can be considered as a family of nested CSs, a TFS can be represented by a family of nested TSs where each TS is regarded as an interval of CSs. Taking into account the α dimension, a TFS can be considered as an interval where its lower and upper bounds are T1FSs. Furthermore, as a TFS is represented by two lower and upper T1FSs under the constraint that the lower T1FS \subset the upper T1FS, it implies that a TFS could be regarded as a T2FS.

From methodological perspectives, TFSs and T2FSs are two facets for dealing with uncertain T1FSs. Therefore, the T2FS and the TFS concepts are not in opposition but rather complementary. If the T2FS representation is usually handled using a functional approach (based on membership functions formalism), the TFS is a set-method which is based on the stacking of TSs according to the α -cuts principle. Furthermore, if the T2FS can be considered as more general than the TFS view, the latter offers the possibility of expressing an uncertain T1FS as an interval of T1FSs. This TFS view allows the manipulation of uncertain T1FSs thanks to the mathematical arsenal of IA (tools and solvers).

The motivation of this paper is to propose an alternative representation of uncertain T1FSs using TFSs. This alternative provides a new outlook on uncertain T1FSs and their manipulation with new computational methodologies. In this context, all the computations will benefit in their implementation from the flexibility and the rigor of IA and interval reasoning.

This paper is organized as follows: Section II describes CSs and TSs. In section III, T1FSs and TFSs are detailed. Combination mechanisms of TSs and TFSs are given in section IV. Potential applications of the TFS approach in the fields of solving linear fuzzy equations and parameters estimation for a nonlinear regressive model have been investigated. In addition, a real-world application for modeling the navigation zone explored by an underwater robot is given. Remarks

and discussions on the pertinence and the applicability of TFSs are proposed in section VI. Finally, conclusions and perspectives are given in section VII.

II. CRISP SETS AND THICK SETS

A. Crisp sets

A subset \mathbb{X} of \mathcal{R}^n (often called a set for simplicity) is defined by the union of singletons $\mathbf{x} = (x_1, \dots, x_n)$ it contains. In this work, conventional sets are named CSs for distinguishing them from other extensions, such as TSs, T1FSs, TFSs and T2FSs.

The characteristic function of a CS \mathbb{X} is defined as follows:

$$\mu_{\mathbb{X}} : \mathcal{R}^n \rightarrow \{0, 1\}; \mathbf{x} \mapsto \begin{cases} 1 & \text{if } \mathbf{x} \in \mathbb{X} \\ 0 & \text{if } \mathbf{x} \notin \mathbb{X} \end{cases} \quad (1)$$

Some standard operations between two CSs \mathbb{X} and \mathbb{Y} , such as the intersection, union and difference, are defined by:

$$\text{Intersection: } \mathbb{X} \cap \mathbb{Y} = \{\mathbf{a} \mid \mathbf{a} \in \mathbb{X} \wedge \mathbf{a} \in \mathbb{Y}\}$$

$$\text{Union: } \mathbb{X} \cup \mathbb{Y} = \{\mathbf{a} \mid \mathbf{a} \in \mathbb{X} \vee \mathbf{a} \in \mathbb{Y}\}$$

$$\text{Difference: } \mathbb{X} \setminus \mathbb{Y} = \{\mathbf{a} \mid \mathbf{a} \in \mathbb{X} \wedge \mathbf{a} \notin \mathbb{Y}\} = \mathbb{X} \cap \bar{\mathbb{Y}}$$

where \wedge and \vee refer to the logical «and» and «or» operators, respectively, and $\bar{\mathbb{Y}}$ is the complement of \mathbb{Y} . Furthermore, inclusion and equality are defined as follows:

$$\text{Inclusion: } \mathbb{X} \subset \mathbb{Y} \Leftrightarrow \forall \mathbf{a} \in \mathbb{X}, \mathbf{a} \in \mathbb{Y}$$

$$\text{Equality: } \mathbb{X} = \mathbb{Y} \Leftrightarrow (\mathbb{X} \subset \mathbb{Y} \wedge \mathbb{Y} \subset \mathbb{X})$$

Due to the simplicity of their computer coding, intervals and boxes are often used as an abstract approximation of CSs. A conventional interval $[a]$ which is denoted by the conventional bracket notation: $[a] = [a^{\text{inf}}, a^{\text{sup}}]$ is a 1D CS, —i.e., a special case of a CS. An interval vector \mathbf{a} is a particular case of a CS of \mathcal{R}^n . In this paper, an interval vector is called a box and it is defined by the Cartesian product of n closed intervals, i.e.,

$$\mathbf{a} = [a_1] \times [a_2] \times \dots \times [a_n]; [a_i] = [a_i^{\text{inf}}, a_i^{\text{sup}}]; \text{ for } i = 1, \dots, n \quad (2)$$

For instance, a 2D box is illustrated in Fig. 2.

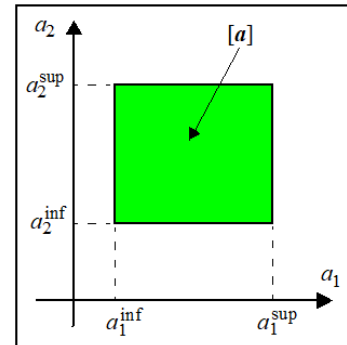


Fig. 2: The representation of the 2D box \mathbf{a}

B. Thick sets (TSs)

Classically, the boundaries of a set are crisp and known with certainty. However, in some situations, the boundaries of sets cannot be sharply described or are hidden for confidentiality purposes. To challenge this problem, the concept of TS has been proposed by Desrochers and Jaulin [30][31]. As illustrated in Fig. 1, a TS $[\mathbb{X}]$ is represented by two CSs \mathbb{X}^{inf} and \mathbb{X}^{sup} , which are regarded as lower and higher bounds of \mathbb{X} . Several different semantics can be associated with the

CSs \mathbb{X}^{inf} and \mathbb{X}^{sup} according to the used domain of application. For example, \mathbb{X}^{inf} and \mathbb{X}^{sup} can be interpreted as the higher and lower bounds of uncertainty in some decision problems, respectively. They can also be interpreted as representations of extreme situations in adverse and favorable situations, respectively. The semantics employed in this paper interprets the CS \mathbb{X}^{inf} as the set of certain elements. The CS \mathbb{X}^{sup} is regarded as a set of plausible elements. The difference $\mathbb{X}^? = \mathbb{X}^{\text{sup}} \setminus \mathbb{X}^{\text{inf}}$ is interpreted as the set of ignorance with plausible but not certain elements.

When denoting by $(\mathbb{P}(\mathfrak{R}^n), \subset)$ the power set of \mathfrak{R}^n equipped with the inclusion order relation \subset , $\mathbb{P}(\mathfrak{R}^n)$ is a complete lattice with respect to \subset (refer to [30][31] for more details). In this context, a TS $\llbracket \mathbb{X} \rrbracket$ of \mathfrak{R}^n is an interval of $(\mathbb{P}(\mathfrak{R}^n), \subset)$ (refer to Fig. 3). Therefore, if $\llbracket \mathbb{X} \rrbracket$ is a TS, there exist two CSs of \mathfrak{R}^n baptized lower bound and upper bound such that:

$$\llbracket \mathbb{X} \rrbracket = [\mathbb{X}^{\text{inf}}, \mathbb{X}^{\text{sup}}] = \{ \mathbb{X} \in \mathbb{P}(\mathfrak{R}^n) \mid \mathbb{X}^{\text{inf}} \subset \mathbb{X} \subset \mathbb{X}^{\text{sup}} \} \quad (3)$$

The TS $\llbracket \mathbb{X} \rrbracket$ is a sublattice of $(\mathbb{P}(\mathfrak{R}^n), \subset)$ [30][31]. Therefore, if $\mathbb{A} \in \llbracket \mathbb{X} \rrbracket$ and $\mathbb{B} \in \llbracket \mathbb{X} \rrbracket$, then $\mathbb{A} \cap \mathbb{B} \in \llbracket \mathbb{X} \rrbracket$ and $\mathbb{A} \cup \mathbb{B} \in \llbracket \mathbb{X} \rrbracket$. If $\mathbb{X}^{\text{inf}} = \mathbb{X}^{\text{sup}} = \mathbb{X}$, $\llbracket \mathbb{X} \rrbracket$ is a CS of \mathfrak{R}^n —i.e., a singleton in $\mathbb{P}(\mathfrak{R}^n)$.

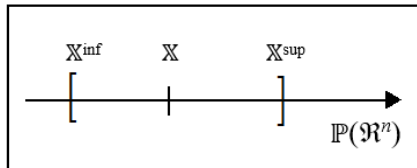


Fig. 3: The representation of the thick set (TS) $\llbracket \mathbb{X} \rrbracket$ as an interval of CSs

In CSs, only the two logic values 0 and 1 are used. In the TS representation, three logic values are necessary: 0 (False), ? (Perhaps) and 1 (True). The fundamental logical operations such as «and», «or» and «not» can be implemented. For instance, it obtains:

$$(? \vee 0) \wedge (? \vee 1) = (? \wedge 1) = ?$$

By analogy with a CS, the characteristic function of a TS $\llbracket \mathbb{X} \rrbracket$ can be defined as follows:

$$\mu_{\llbracket \mathbb{X} \rrbracket} : \mathfrak{R}^n \rightarrow \{0, ?, 1\}; \mathbf{x} \mapsto \begin{cases} 1 & \text{if } \mathbf{x} \in \mathbb{X}^{\text{inf}} \\ 0 & \text{if } \mathbf{x} \notin \mathbb{X}^{\text{sup}} \\ ? & \text{otherwise} \end{cases} \quad (4)$$

When two TSs $\llbracket \mathbb{X} \rrbracket$ and $\llbracket \mathbb{Y} \rrbracket$ are considered, the arithmetical and logical operators between CSs can be extended to TSs [30][31]. This point will be discussed later in the paper.

A thick interval (TI) $\llbracket [a] \rrbracket = \llbracket [a^{\text{inf}}, a^{\text{sup}}] \rrbracket$, $[a^{\text{inf}}] \subset [a^{\text{sup}}]$, is a 1D TS, —i.e., a special case of a TS (see Fig. 4). If $[a^{\text{inf}}] = [a^{\text{sup}}]$, the TI $\llbracket [a] \rrbracket$ becomes a conventional interval.

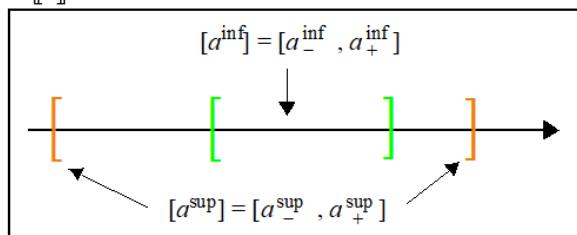


Fig. 4: The representation of the thick interval (TI) $\llbracket [a] \rrbracket$

A TI vector $\llbracket [a] \rrbracket = \llbracket [a^{\text{inf}}, a^{\text{sup}}] \rrbracket$ which is called a thick box (TB) is a particular case of a TS of $\mathbb{P}(\mathfrak{R}^n)$. Since the bounds

$[a^{\text{inf}}]$ and $[a^{\text{sup}}]$ are boxes of \mathfrak{R}^n , they can be expressed as the Cartesian product of n intervals:

$$[a^{\text{inf}}] = [a_1^{\text{inf}}] \times [a_2^{\text{inf}}] \times \dots \times [a_n^{\text{inf}}]; [a^{\text{sup}}] = [a_1^{\text{sup}}] \times [a_2^{\text{sup}}] \times \dots \times [a_n^{\text{sup}}]$$

For example, a 2D TB is illustrated in Fig. 5.

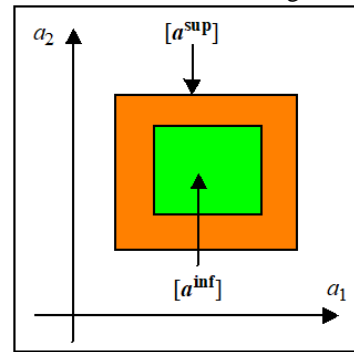


Fig. 5: The representation of the 2D thick box (TB) $\llbracket [a] \rrbracket$

C. Application example 1

The objective of this example is to show how a TS is modeled. Let us consider a robot R that is moving on a two-dimensional path and is at position $\mathbf{x} = (x_1, x_2)$. For its navigation, the robot R needs to communicate with a WiFi hotspot W . The position of W is located at $\mathbf{p} = (p_1, p_2)$. The robot receiver provides that if R is at a distance less than 35m to W (the distance of detection), the communication between R and W is possible (see Fig. 6). Therefore, when the position of W can exactly be determined, the robot navigation zone is a CS \mathbb{Z} , which is defined by:

$$\mathbb{Z} = \{ \mathbf{x} = (x_1, x_2) \in \mathfrak{R}^2 \mid \|\mathbf{x} - \mathbf{p}\| \leq 35 \} \quad (5)$$

It can be stated that the CS \mathbb{Z} is simply the area of a circle of center \mathbf{p} and radius 35.

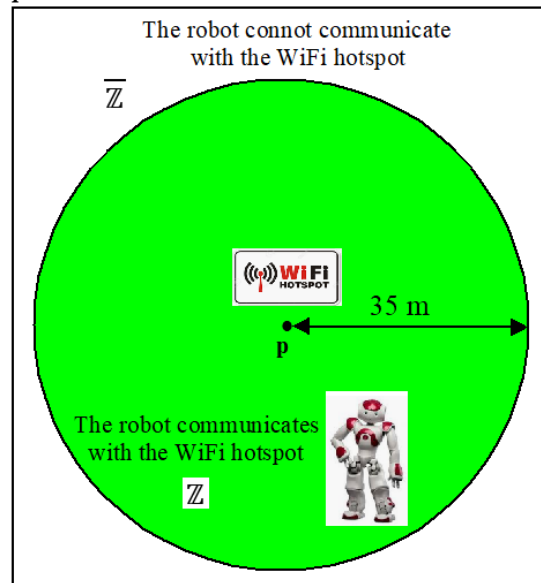


Fig. 6: The crisp set \mathbb{Z} representing the robot navigation zone

Due to the presence of disturbances, the position of W cannot be situated precisely. Let us assume that the only information available is that W is located in a 2D box $\llbracket [p] \rrbracket$ (see Fig. 7). Our objective now is to model the robot navigation zone according to the uncertainties induced by the position of W . Due to the uncertain position of W , two nested CSs (navigation zones)

\mathbb{Z}^{inf} and \mathbb{Z}^{sup} can be envisioned. They are computed according to the following expressions:

$$\mathbb{Z}^{\text{inf}} = \{\mathbf{x} = (x_1, x_2), \forall \mathbf{p} = (p_1, p_2) \in [\mathbf{p}] \mid \|\mathbf{x} - \mathbf{p}\| \leq 35\} \quad (6)$$

$$\mathbb{Z}^{\text{sup}} = \{\mathbf{x} = (x_1, x_2), \exists \mathbf{p} = (p_1, p_2) \in [\mathbf{p}] \mid \|\mathbf{x} - \mathbf{p}\| \leq 35\} \quad (7)$$

The CS \mathbb{Z}^{inf} refers to the zone where the communication between R and W is always certain regardless of the WiFi hotspot position \mathbf{p} in $[\mathbf{p}]$. Hence, $\forall \mathbf{p} = (p_1, p_2) \in [\mathbf{p}]$, the robot is in the zone \mathbb{Z}^{inf} delimited by the constraint $\|\mathbf{x} - \mathbf{p}\| \leq 35$ where the communication between R and W is always certain. The CS \mathbb{Z}^{sup} , encompassing \mathbb{Z}^{inf} , corresponds to the zone where the communication between R and W is plausible—i.e., there is at least one position \mathbf{p} in $[\mathbf{p}]$ ($\exists \mathbf{p} = (p_1, p_2) \in [\mathbf{p}]$) such as $\|\mathbf{x} - \mathbf{p}\| \leq 35$. Therefore, \mathbb{Z}^{inf} and \mathbb{Z}^{sup} represent certain and plausible navigation zones, respectively.

In this framework, the robot navigation zone is no longer a CS but an uncertain set such as $\mathbb{Z}^{\text{inf}} \subset \mathbb{Z} \subset \mathbb{Z}^{\text{sup}}$ and can be expressed by the TS: $[\mathbb{Z}] = [\mathbb{Z}^{\text{inf}}, \mathbb{Z}^{\text{sup}}]$. Therefore, the navigation zone becomes uncertain because its boundaries are uncertain. The difference $\mathbb{Z}^? = \mathbb{Z}^{\text{sup}} \setminus \mathbb{Z}^{\text{inf}}$ characterizes the uncertainty and is interpreted as the zone where the communication between R and W is plausible but not certain (perhaps possible). From a practical point of view, the CSs \mathbb{Z}^{sup} and \mathbb{Z}^{inf} are implemented as follows:

$$\begin{aligned} \mathbb{Z}^{\text{sup}} &= \{\mathbf{x} = (x_1, x_2), \exists \mathbf{p} = (p_1, p_2) \in [\mathbf{p}] \mid \|\mathbf{x} - \mathbf{p}\| \leq 35\} \quad (8) \\ &= \text{proj}_x \{(x_1, x_2, p_1, p_2) \mid (x_1 - p_1)^2 + (x_2 - p_2)^2 \leq 1225\} \end{aligned}$$

$$\begin{aligned} \mathbb{Z}^{\text{inf}} &= \{\mathbf{x} = (x_1, x_2), \forall \mathbf{p} = (p_1, p_2) \in [\mathbf{p}] \mid \|\mathbf{x} - \mathbf{p}\| \leq 35\} \\ &= \overline{\{\mathbf{x} = (x_1, x_2), \exists \mathbf{p} = (p_1, p_2) \in [\mathbf{p}] \mid \|\mathbf{x} - \mathbf{p}\| > 35\}} \\ &= \text{proj}_x \{(x_1, x_2, p_1, p_2) \mid (x_1 - p_1)^2 + (x_2 - p_2)^2 > 1225\} \end{aligned} \quad (9)$$

where $(x_1, x_2) \in \mathbb{R}^2$ and $(x_1, x_2, p_1, p_2) \in \mathbb{R}^2 \times [\mathbf{p}]$.

In (8) and (9), proj_x refers to a projection operation on the x -space. This projection operation is implemented using the method proposed in [32] (see the Appendix for its principle).

In equations (8)-(9), the formulation with the intersection, projection, Cartesian product, and complement operations aim to facilitate the implementation using interval-based solvers such as PyIbex (see benensta.github.io/pyIbex/). All the results presented in the paper are obtained using the PyIbex solver. All the figures were built using the visualization system VIBE (see enstabretagnerobotics.github.io/VIBE/).

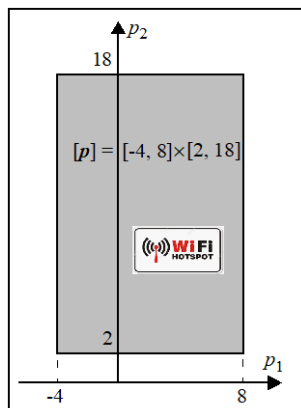


Fig. 7: A 2D box $[\mathbf{p}]$ representing the WiFi hotspot position

For instance, the results obtained using the 2D box $[\mathbf{p}] = [-4, 8] \times [2, 18]$ of Fig. 7 are shown in Figs. 8-9 with and without paving illustration (see the Appendix for this implementation using PyIbex).

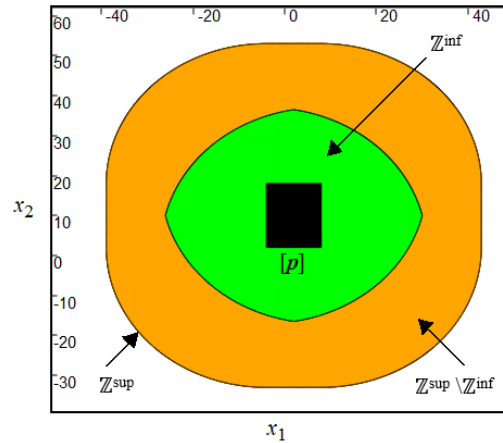


Fig. 8: The TS (navigation zone) $[\mathbb{Z}] = [\mathbb{Z}^{\text{inf}}, \mathbb{Z}^{\text{sup}}]$ without paving

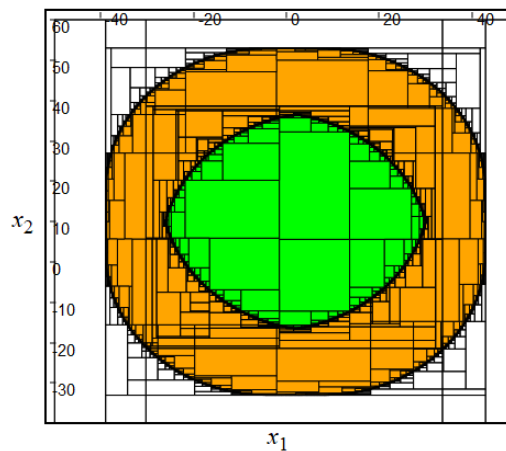


Fig. 9: The TS (navigation zone) $[\mathbb{Z}] = [\mathbb{Z}^{\text{inf}}, \mathbb{Z}^{\text{sup}}]$ with paving

All the results were obtained using the paving procedure given in [33]. For reasons of visibility, in the paper sequel, the paving is not illustrated in the figures.

III. TYPE-1 FUZZY SETS AND THICK FUZZY SETS

A. Type-1 fuzzy sets (T1FSs)

A T1FS can be represented either by a membership function or by a collection of α -cuts. For its compatibility with the set and interval-based approaches, the α -cuts method is adopted in our method. Therefore, according to the α -cuts principle, a T1FS can be decomposed into a weighted combination of CSs using the representation theorem [25][34]. This concept has been used as the foundation for generalizing concepts and methods based on CSs into those based on T1FSs.

Let \mathbb{U} be a CS called universe. The representation theorem states that any T1FS can be decomposed into a system (a family denoted in bold) $\mathbf{X} = \{\mathbb{X}(\alpha)\}_\alpha; \alpha \in [0, 1]$ of its α -cuts in \mathbb{U} under the constraint of monotonicity (consistency)—i.e., if $\alpha_1 > \alpha_2$ then $\mathbb{X}(\alpha_1) \subseteq \mathbb{X}(\alpha_2)$. Therefore, for a given $\alpha \in [0, 1]$, an α -cut of a T1FS is a CS $\mathbb{X}(\alpha)$. Furthermore, for any nested system of α -cuts: $\mathbf{X} = \{\mathbb{X}(\alpha)\}_\alpha; \alpha \in [0, 1]$, a T1FS with a membership function $\mu_{\mathbf{X}}: \mathbb{U} \rightarrow [0, 1]$ can be constructed. In

the paper reminder, a T1FS which is denoted \mathbf{X} (in bold) is considered as a family of α -cuts and is defined as follows:

$$\mathbf{X} = \bigcup_{\alpha \in [0,1]} \alpha \mathbf{X}(\alpha) \quad (10)$$

The α -cut $\mathbf{X}(\alpha)$ is a CS. Thanks' to the α -cut principle, all the operations on T1FSs are performed as operations between CSs. An interval-valued T1FS (IV-T1FS) is a 1D T1FS—i.e., a special case of a T1FS. For compatibility with the interval notation, an IV-T1FS is denoted as follows:

$$[A] = \bigcup_{\alpha \in [0,1]} \alpha [A(\alpha)] \quad (11)$$

Where $A(\alpha)$ is an α -cut of $[A]$ and can be expressed by a conventional interval, i.e., $A(\alpha) = [a_\alpha^{\text{inf}}, a_\alpha^{\text{sup}}]$. An IV-T1FS vector $[A]$ is defined by the Cartesian product of n IV-T1FSs. In this paper, an IV-T1FS vector is called a box-valued T1FS (BV-T1FS) and it is defined by the following expression:

$$[A] = [A_1] \times [A_2] \times \dots \times [A_n]$$

At each α -cut level, $[A(\alpha)]$ is a box. An example of a 2D BV-T1FS is illustrated in Fig. 10.

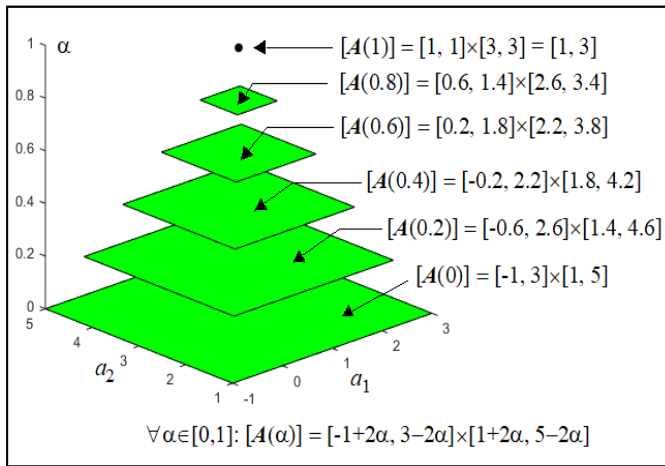


Fig. 10: The representation of a BV-T1FS $[A]$ using its α -cuts

B. Thick fuzzy sets (TFSs)

Similar to T1FSs, a TFS can be defined by a system (a family) of nested TSs: $\llbracket \mathbf{X} \rrbracket = \{ \llbracket \mathbf{X}(\alpha) \rrbracket \}_\alpha; \alpha \in [0,1]$ of its α -cuts under the monotonicity (consistency) constraint: if $\alpha_1 > \alpha_2$ then $\llbracket \mathbf{X}(\alpha_1) \rrbracket \subseteq \llbracket \mathbf{X}(\alpha_2) \rrbracket$. Therefore, a TFS which is considered as a family of α -cuts and is defined as follows:

$$\begin{aligned} \llbracket \mathbf{X} \rrbracket &= \llbracket \mathbf{X}^{\text{inf}}, \mathbf{X}^{\text{sup}} \rrbracket = \bigcup_{\alpha \in [0,1]} \alpha \llbracket \mathbf{X}(\alpha) \rrbracket \quad (12) \\ &= \llbracket \bigcup_{\alpha \in [0,1]} \alpha \mathbf{X}^{\text{inf}}(\alpha), \bigcup_{\alpha \in [0,1]} \alpha \mathbf{X}^{\text{sup}}(\alpha) \rrbracket \end{aligned}$$

where the α -cut $\llbracket \mathbf{X}(\alpha) \rrbracket$ is a TS. From (12) and according to the vertical dimension α , it can be stated that \mathbf{X}^{inf} and \mathbf{X}^{sup} are T1FSs. Therefore, the TFS $\llbracket \mathbf{X} \rrbracket$ is represented by an interval of T1FSs where \mathbf{X}^{inf} and \mathbf{X}^{sup} represent its bounds. In the situation when $\mathbf{X}^{\text{inf}} = \mathbf{X}^{\text{sup}} = \mathbf{X}$, the TFS $\llbracket \mathbf{X} \rrbracket$ becomes a T1FS \mathbf{X} . In this framework, as the TFS is composed of two T1FSs (lower \mathbf{X}^{inf} and upper \mathbf{X}^{sup} T1FSs) under the constraint $\mathbf{X}^{\text{inf}} \subset \mathbf{X}^{\text{sup}}$, it implies that a TFS could be regarded as a T2FS. The lower bound \mathbf{X}^{inf} represents a T1FS which is certain. The T1FS bound \mathbf{X}^{sup} is an upper bound which delimits all the T1FSs

that are plausible. The uncertainty is exhibited by the penumbra $\mathbf{X}^{\text{sup}} \setminus \mathbf{X}^{\text{inf}}$. According to the α -cut dimension, the penumbra concept in the TFS representation could be regarded as the FOU phenomenon in T2FS representation.

Due to the computational complexity of T2FSs, several works have focused on interval-valued T2FSs (IV-T2FSs). Thus, an IV-T2FS has the advantage of simplicity in computer implementation. An IV-T2FS is a 1D T2FS—i.e., a special case of a T2FS (refer to Fig. 11 for a IV-T2FS illustration). A TFS can represent an IV-T2FS as follows:

$$\begin{aligned} \llbracket A \rrbracket &= \bigcup_{\alpha \in [0,1]} \alpha \llbracket A(\alpha) \rrbracket = \llbracket [A^{\text{inf}}], [A^{\text{sup}}] \rrbracket \quad (13) \\ &= \llbracket \bigcup_{\alpha \in [0,1]} \alpha [A^{\text{inf}}(\alpha)], \bigcup_{\alpha \in [0,1]} \alpha [A^{\text{sup}}(\alpha)] \rrbracket \end{aligned}$$

where $\llbracket A(\alpha) \rrbracket = \llbracket [A^{\text{inf}}(\alpha)], [A^{\text{sup}}(\alpha)] \rrbracket$ is an α -cut of $\llbracket A \rrbracket$ and is regarded as a TI. An IV-T2FS vector is called here a box-valued T2FS (BV-T2FS). Since the fuzzy boundaries $[A^{\text{inf}}$ and $[A^{\text{sup}}]$ are BV-T1FSs it obtains:

$$\begin{aligned} [A^{\text{inf}}] &= [A_1^{\text{inf}}] \times [A_2^{\text{inf}}] \times \dots \times [A_n^{\text{inf}}] \\ [A^{\text{sup}}] &= [A_1^{\text{sup}}] \times [A_2^{\text{sup}}] \times \dots \times [A_n^{\text{sup}}] \end{aligned}$$

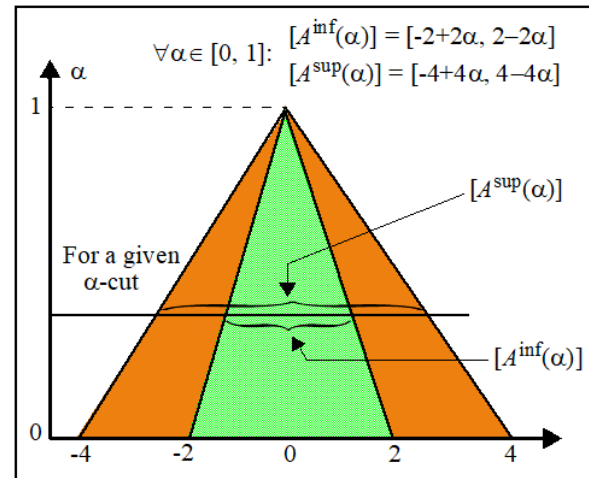


Fig. 11: The representation of an IV-T2FS (1D-TFS)

For example, a 2D IV-T2FS is illustrated in Fig. 12. A TFS can represent a BV-T2FS as follows:

$$\llbracket A \rrbracket = \llbracket [A^{\text{inf}}], [A^{\text{sup}}] \rrbracket$$

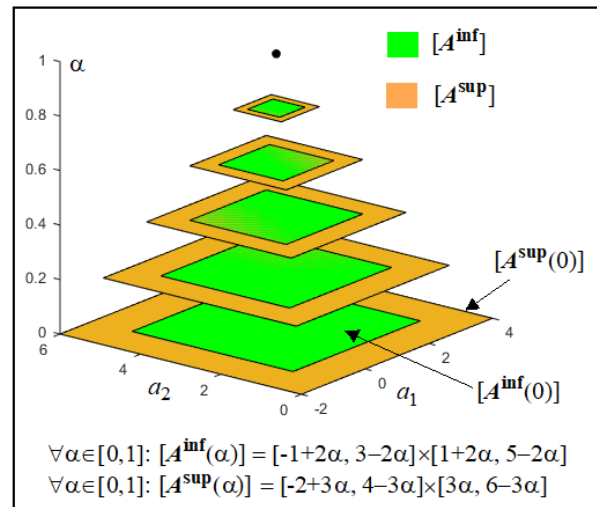


Fig. 12: The representation of a BV-T2FS $\llbracket A \rrbracket$ using its α -cuts

C. Application example 2

This application aims at demonstrating how a TFS is built for modeling a robot navigation zone. Let us reconsider the example of application 1. The position of the WiFi hotspot W is still uncertain and is given by the BV-TIFS $[P]$ which is defined by its α -cuts as follows:

$$\forall \alpha \in [0, 1]: [P(\alpha)] = [-4+6\alpha, 8-6\alpha] \times [2+8\alpha, 18-8\alpha]$$

Therefore, the box $[P(0)] = [-4, 8] \times [2, 18]$ corresponds to the most uncertain location of W and $[P(1)] = [2, 2] \times [10, 10] = (2, 10)$ refers to its most precise position. At each level α on $[P(\alpha)]$, a degree of confidence $(1-\alpha)$ can be associated. These confidence levels may depend on the weather and some environmental factors. The BV-TIFS $[P(\alpha)]$ is illustrated in Fig. 13 for six values of $\alpha = 0, 0.2, 0.4, 0.6, 0.8$ and 1.

In the situation where the position of W is uncertain and in order to be able to detect its signal, the robot R can ameliorate the sensitivity of its receivers to increase the detection distance. For simplicity of illustration, we consider that the relation between the detection distance of R and the level of uncertainty in the localization of W is given by the linear relation $35-15\alpha$.

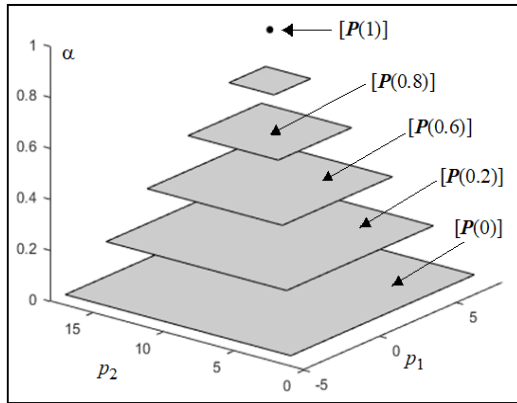


Fig. 13: The representation of the BV-TIFS $[P]$ using its α -cuts

Therefore, $\alpha=0$ means that if R is at a distance less than 35 m to W , it can communicate with W . Also, $\alpha=1$ stipulate that the communication between R and W is possible if the distance between them is less than 20 m.

By applying the concept of TSs, the results obtained for the α -cuts $\alpha = 0, 0.5$ and 1 are shown in Figs. 14-16.

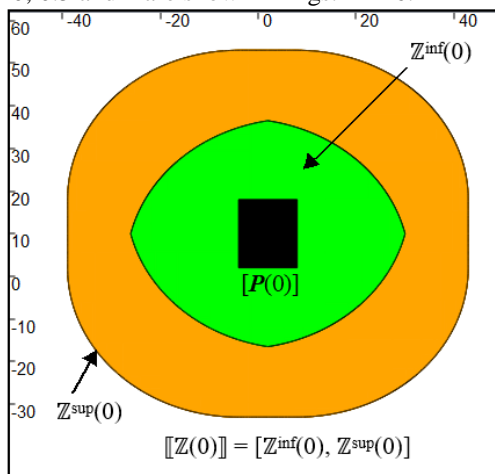


Fig. 14: The robot navigation zone $[Z(0)]$

At each α -cut level, the WiFi hotspot W is in a box $[P(\alpha)]$ and the corresponding navigation zone is a TS represented by $[Z(\alpha)] = [Z^{inf}(\alpha), Z^{sup}(\alpha)]$.

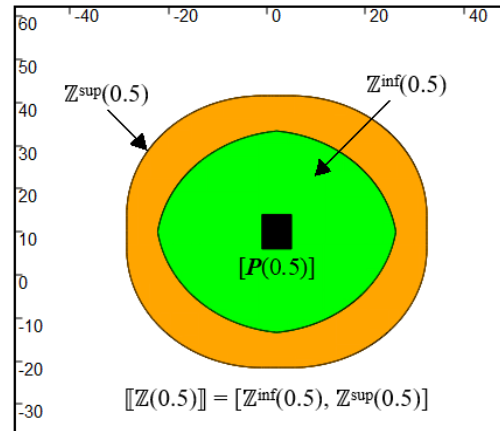


Fig. 15: The robot navigation zone $[Z(0.5)]$

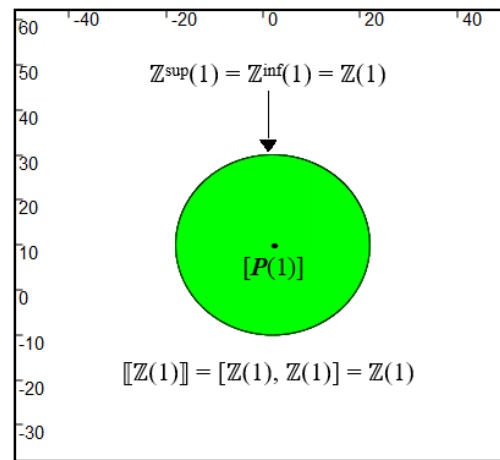


Fig. 16: The robot navigation zone $[Z(1)]$

The concatenation of the three α -cut levels for $Z^{inf}(\alpha)$ and $Z^{sup}(\alpha)$ leads to Figs. 17-18. The colors in Figs. 17-18 are only used to differentiate the α -cuts.

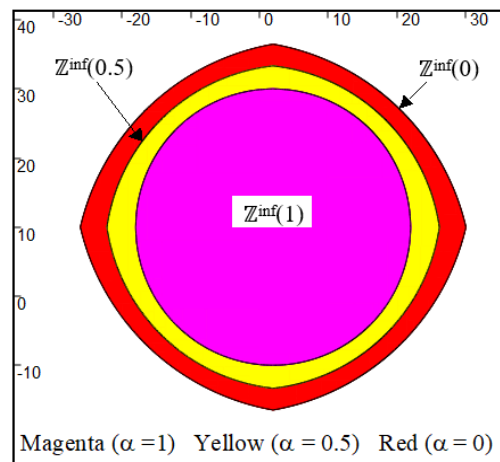


Fig. 17: The lower bounds $Z^{inf}(0)$, $Z^{inf}(0.5)$ and $Z^{inf}(1)$

According to these results, it can be stated that, if the position of W is uncertain, the navigation zone is a TS. On the opposite, if the position of W is crisp, the $[Z(\alpha)]$ becomes a CS (see Fig. 16)—i.e., $Z^{sup}(1) = Z^{inf}(1) = Z(1)$. In this application, it can be checked that (see Figs. 17-18):

$Z^{\text{inf}}(1) \subset Z^{\text{inf}}(0.5) \subset Z^{\text{inf}}(0)$; and: $Z^{\text{sup}}(1) \subset Z^{\text{sup}}(0.5) \subset Z^{\text{sup}}(0)$

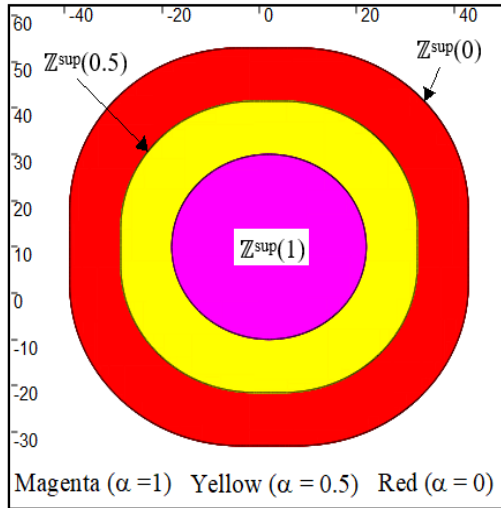


Fig. 18: The upper bounds $Z^{\text{sup}}(0)$, $Z^{\text{sup}}(0.5)$ and $Z^{\text{sup}}(1)$

More generally, it can be verified that:

$$\forall \alpha_1, \alpha_2 \in [0, 1]: \alpha_1 \leq \alpha_2 \Leftrightarrow$$

$$Z^{\text{sup}}(\alpha_2) \subset Z^{\text{sup}}(\alpha_1); \text{ and: } Z^{\text{sup}}(\alpha_2) \subset Z^{\text{sup}}(\alpha_1);$$

For each α -cut, $Z^{\text{sup}}(\alpha)$ and $Z^{\text{inf}}(\alpha)$ are respectively the lower and upper bounds of a TS $[[Z(\alpha)]]$. The navigation zone is a family of nested TSs, thus elaborating a TFS. Furthermore, for $\alpha \in [0, 1]$, the CSs $Z^{\text{inf}}(\alpha)$ forms a family of nested CSs. The same remark is valid for the CSs $Z^{\text{sup}}(\alpha)$. Therefore, these families of nested CSs can be regarded as T1FSs.

IV. COMBINATION OF TSS AND TFSS

A. Combination principle

Knowing that at each α -cut a TFS is a TS, operations on TFSs are performed using operations on TSs. Let us consider a collection of CSs $\{X_i\}_{i \in \Omega}$. The smallest TS which encloses all $\{X_i\}_{i \in \Omega}$ is defined as follows:

$$\textcircled{S}\{X_i, i \in \Omega\} = [[\bigcap_{i \in \Omega} X, \bigcup_{i \in \Omega} X]]$$

where \textcircled{S} denotes the smallest TS. In this context, it is possible to extend the operators initially proposed for CSs to thick sets as follows [30][31]:

$$[[X]] \bullet [[Y]] = \textcircled{S}\{T, \exists X \in [[X]], \exists Y \in [[Y]], T = X \bullet Y\};$$

$$\bullet \in \{\cap, \cup, \setminus, \dots\}$$

For instance, according to the monotony property of the intersection, union, difference and addition operators, they are defined by the following expressions:

$$\text{Intersection: } [[X]] \cap [[Y]] = [[X^{\text{inf}} \cap Y^{\text{inf}}, X^{\text{sup}} \cap Y^{\text{sup}}]]$$

$$\text{Union: } [[X]] \cup [[Y]] = [[X^{\text{inf}} \cup Y^{\text{inf}}, X^{\text{sup}} \cup Y^{\text{sup}}]]$$

$$\text{Difference: } [[X]] \setminus [[Y]] = [[X^{\text{inf}} \setminus Y^{\text{sup}}, X^{\text{sup}} \setminus Y^{\text{inf}}]]$$

$$\text{Addition: } [[X]] + [[Y]] = [[X^{\text{inf}} + Y^{\text{inf}}, X^{\text{sup}} + Y^{\text{sup}}]]$$

More generally, for a given function f from \mathfrak{R}^n to \mathfrak{R}^m , the image of the TS $[[X]] = [[X^{\text{inf}}, X^{\text{sup}}]]$ by f is evaluated by the following expression:

$$f([[X]]) = \textcircled{S}\{f(X), X \in [[X]]\} = [[f(X^{\text{inf}})], [f(X^{\text{sup}})]]$$

This extension of functions allows the propagation of TSs and TFSs through linear and nonlinear models and controllers where the inputs, outputs, states and/or parameters can be

represented by TFSs (or their simplified versions such as IV-T1FSs, BV-T1FSs, IV-T2FS and BV-T2FSs).

B. Application example 3

The objective of this application is to show the TSs combination mechanism. For concision reasons, only the operators \cap and \cup are illustrated on the 0-cut. Other operators can be implemented using the same methodology and for any α -cut level. The application example 2 is reconsidered here where 4 WiFi hotspots W_i , $i=1, \dots, 4$ are assumed to be available for helping the robot R in its navigation. The position of each hotspot W_i is given by a BV-T1FS. These BV-T1FSs are given by their α -cut representation as follows:

$$[P_1(\alpha)] = [-4+6\alpha, 8-6\alpha] \times [2+8\alpha, 18-8\alpha]$$

$$[P_2(\alpha)] = [12+4\alpha, 20-4\alpha] \times [-4+5\alpha, 7-6\alpha]$$

$$[P_3(\alpha)] = [10+5\alpha, 20-5\alpha] \times [20+5\alpha, 30-5\alpha]$$

$$[P_4(\alpha)] = [-9+2\alpha, -5-2\alpha] \times [-10+2\alpha, -1-7\alpha]$$

For $\alpha = 0$, the positions of the hotspots are 2D boxes and are illustrated in Fig. 19. Therefore, for each hotspot W_i in its box $[P_i(0)]$, if the robot R is located at a distance less than 35m, the communication between them is possible. In this case, the navigation zone relative to each hotspot is a TS given by:

$$[[Z_i(0)]] = [[Z_i^{\text{inf}}(0), Z_i^{\text{sup}}(0)]]; \text{ for } i = 1, \dots, 4$$

Where $Z_i^{\text{inf}}(0)$ and $Z_i^{\text{sup}}(0)$ are CSs. For example, upper bounds $Z_i^{\text{sup}}(0), i=1, \dots, 4$ are illustrated in Fig. 20 where the colors are used only to differentiate the CSs. Moreover, the CSs are not filled for visibility reasons.

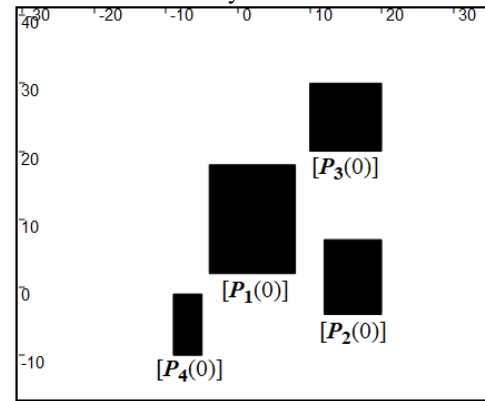


Fig. 19: The representation of the boxes $[P_i(0)], i = 1, \dots, 4$

In this application, two navigation strategies can be considered. The first is considered as the most reliable but the most pessimistic where the navigation requires the contribution of all the hotspots. In the optimistic situation (the second strategy) which is regarded as the least reliable, the contribution of at least one hotspot is sufficient for navigation. Therefore, the pessimistic navigation zone $[[Z_{\cap}(0)]]$ and the optimistic navigation zone $Z_{\cup}(0)$ are illustrated in Figs. 21-22 and can be formulated as follows:

$$[[Z_{\cap}(0)]] = [[Z_{\cap}^{\text{inf}}(0), Z_{\cap}^{\text{sup}}(0)]] =$$

$$[[\bigcap_{i=1, \dots, 4} Z_i^{\text{inf}}(0), \bigcap_{i=1, \dots, 4} Z_i^{\text{sup}}(0)]]; \text{ and:}$$

$$[[Z_{\cup}(0)]] = [[Z_{\cup}^{\text{inf}}(0), Z_{\cup}^{\text{sup}}(0)]] =$$

$$[[\bigcup_{i=1, \dots, 4} Z_i^{\text{inf}}(0), \bigcup_{i=1, \dots, 4} Z_i^{\text{sup}}(0)]]$$

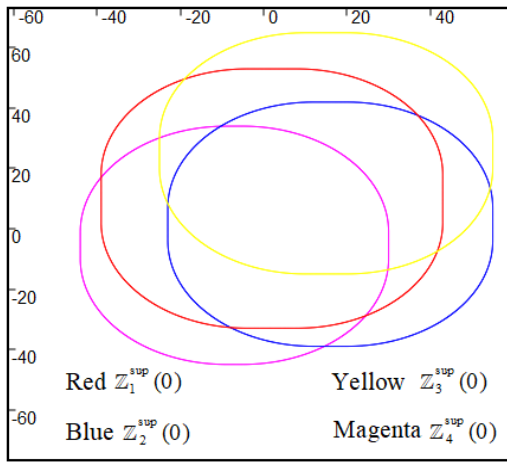


Fig. 20: The upper bounds $Z_i^{\text{sup}}(0), i = 1, \dots, 4$

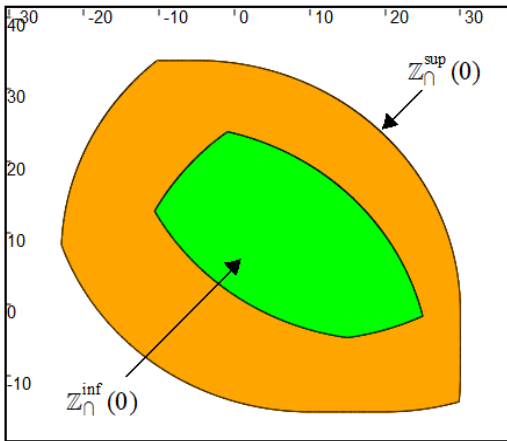


Fig. 21: The representation of the pessimistic TS $\llbracket Z_n(0) \rrbracket$

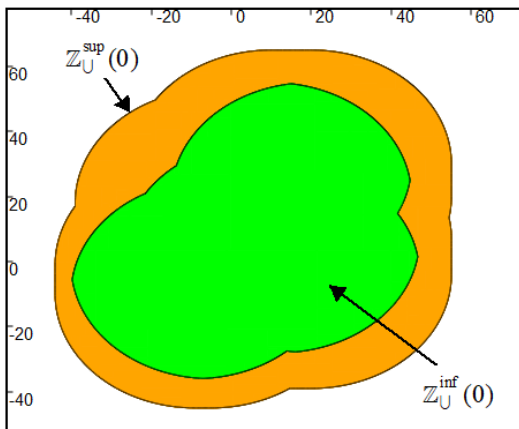


Fig. 22: The representation of the optimistic TS $\llbracket Z_U(0) \rrbracket$

V. POTENTIAL APPLICATIONS OF TFSS

In this section, three applications are presented. Two applications are academic and the third application is a real-world application. The first application is related to the resolution of a system of equations with IV-T1FSs parameters. For simplicity, the equations are taken as linear. However, the approach is applicable regardless of the nonlinear form of the equations. The second application concerns the problem of estimating parameters of a nonlinear regressive parametric model. The objective is to determine all the uncertain

realizable parameters in the situation where the inputs and outputs are represented by IV-T1FSs. The third application focuses on the characterization of an uncertain zone that is explored by an underwater robot.

A. Solving a system of linear fuzzy equations

Let us consider the fuzzy linear system of equations given by the following expression:

$$\begin{cases} [A_{11}(\alpha)] \times x_1 + [A_{12}(\alpha)] \times x_2 \in [C_1(\alpha)] \\ [A_{21}(\alpha)] \times x_1 + [A_{22}(\alpha)] \times x_2 \in [C_2(\alpha)] \end{cases} \quad (14)$$

where the parameters $[A_{ij}(\alpha)]$; $i, j = 1, 2$ are IV-T1FSs:

$$\begin{aligned} [A_{11}(\alpha)] &= [2+\alpha, 4-\alpha]; & [A_{12}(\alpha)] &= [-2+1.5\alpha, 1-1.5\alpha] \\ [A_{21}(\alpha)] &= [-1+1.5\alpha, 2-1.5\alpha]; & [A_{22}(\alpha)] &= [2+\alpha, 4-\alpha] \\ [C_1(\alpha)] &= [-2+2\alpha, 2-2\alpha]; & [C_2(\alpha)] &= [-2+2\alpha, 2-2\alpha] \end{aligned}$$

The system (14) can be written in a matrix form as follows:

$$[A(\alpha)] \times \mathbf{x} \in [C(\alpha)]; \text{ with: } [A(\alpha)] = \begin{pmatrix} [A_{11}(\alpha)] & [A_{12}(\alpha)] \\ [A_{21}(\alpha)] & [A_{22}(\alpha)] \end{pmatrix}; [C(\alpha)] = \begin{pmatrix} [C_1(\alpha)] \\ [C_2(\alpha)] \end{pmatrix}; \mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

For $\alpha = 0$, the system (14) corresponds to a well-known interval linear system which is reconsidered in [35]. The set of solutions of the interval system (14) given in [35] for $\alpha = 0$ is a CS which is illustrated in Fig. 23. The solution in Fig. 23 corresponds to that produced by conventional interval solvers like the Intlab solver (see <http://www.ti3.tu-harburg.de/intlab/>). Using our uncertain approach, for each level α , the solution of (14) is not given by a CS but by a TS $\llbracket \mathbb{X}(\alpha) \rrbracket$, composed of two CSs. A CS $\mathbb{X}^{\text{inf}}(\alpha)$ of certain solutions and a CS $\mathbb{X}^{\text{sup}}(\alpha)$ of plausible solutions. These solutions are depicted in Fig. 24 for $\alpha = 0$ and computed by the following expressions:

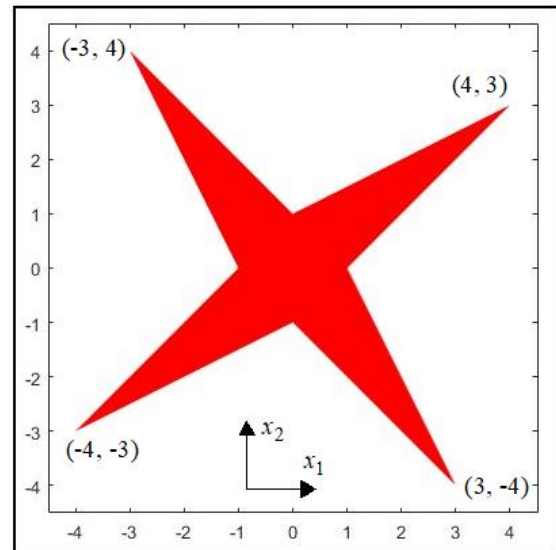


Fig. 23: The CS solution of (14) for $\alpha = 0$ using the conventional IA approach

$$\begin{aligned} \mathbb{X}^{\text{inf}}(\alpha) &= \{ \mathbf{x} = (x_1, x_2), \forall A \in [A(\alpha)] \mid A \times \mathbf{x} \in [C(\alpha)] \} \\ &= \bigcap_{i,j \in \{1,2\}} \{ \mathbf{x} = (x_1, x_2), \forall A_{ij} \in [A_{ij}(\alpha)] \mid A \times \mathbf{x} \in [C(\alpha)] \} \\ &= \bigcap_{i,j \in \{1,2\}} \overline{\{ \mathbf{x} = (x_1, x_2), \exists A_{ij} \in [A_{ij}(\alpha)] \mid A \times \mathbf{x} \notin [C(\alpha)] \}} \\ \mathbb{X}^{\text{sup}}(\alpha) &= \{ \mathbf{x} = (x_1, x_2), \exists A \in [A(\alpha)] \mid A \times \mathbf{x} \in [C(\alpha)] \} \\ &= \bigcap_{i,j \in \{1,2\}} \{ \mathbf{x} = (x_1, x_2), \exists A_{ij} \in [A_{ij}(\alpha)] \mid A \times \mathbf{x} \in [C(\alpha)] \} \end{aligned}$$

The CSs $\mathbb{X}^{\text{inf}}(\alpha)$ and $\mathbb{X}^{\text{sup}}(\alpha)$ are interpreted as follows: the certain CS $\mathbb{X}^{\text{inf}}(\alpha)$ refers to the set of certain solutions x such that whatever the position of the parameters A_{ij} in $[A_{ij}(\alpha)]$ ($\forall A_{ij} \in [A_{ij}(\alpha)]$, $i = 1, 2$ and $j = 1, 2$), the values of x are solutions of (14).

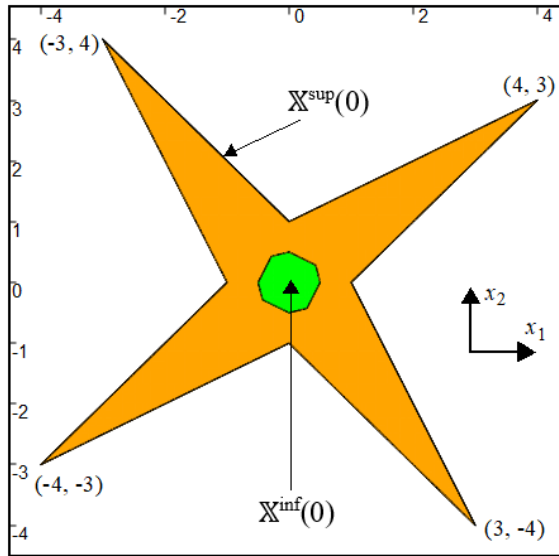


Fig. 24: The TS $\mathbb{X}(0)$: solution of (14) for $\alpha = 0$ using our approach

The plausible CS $\mathbb{X}^{\text{sup}}(\alpha)$ is interpreted as the set of plausible solutions x such that there is at least one set of parameters A_{ij} in $[A_{ij}(\alpha)]$ ($\exists A_{ij} \in [A_{ij}(\alpha)]$, $i = 1, 2$ and $j = 1, 2$) for which the values of x are solutions of (14). By comparing Figs. 23 and 24, it can be stated that the solution set given in [35] or by using Matlab interval solver corresponds to the plausible set $\mathbb{X}^{\text{sup}}(\alpha)$. Therefore, the majority of interval solvers are only interested in plausible solutions. By integrating the dimension $\alpha \in [0, 1]$, the certain and plausible CSs become T1FSs (see Fig. 25 when a sampling step size of 0.1 on α is used).

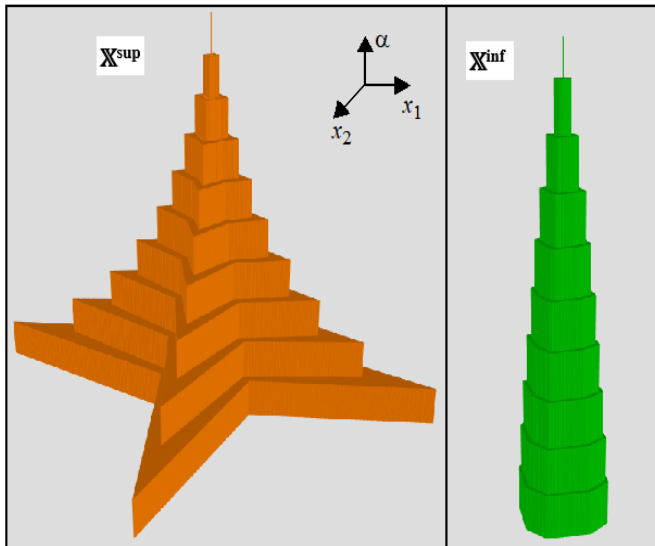


Fig. 25: The T1FSs \mathbb{X}^{sup} and \mathbb{X}^{inf}

The stacking of all the TSs $\mathbb{X}(\alpha) = [\mathbb{X}^{\text{inf}}(\alpha), \mathbb{X}^{\text{sup}}(\alpha)]$ leads to the TFS which is illustrated in Fig. 26. This TFS is composed of two plausible \mathbb{X}^{sup} and certain \mathbb{X}^{inf} T1FSs.

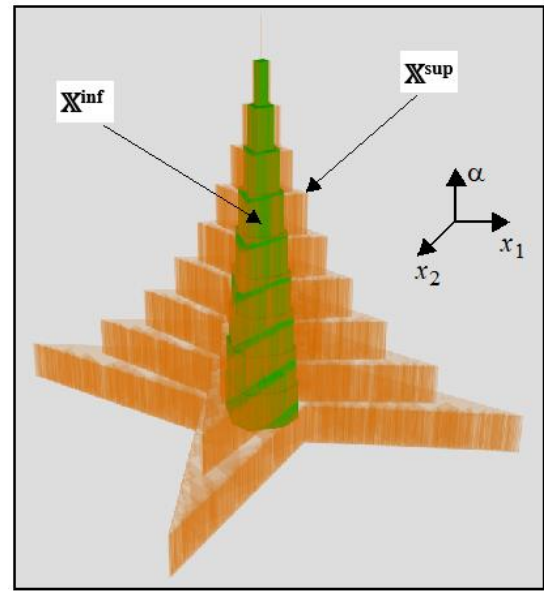


Fig. 26: The TFS $\mathbb{X} = [\mathbb{X}^{\text{inf}}, \mathbb{X}^{\text{sup}}]$

B. Parameters estimation of a nonlinear regressive model

Let us assume a set of fuzzy input/output data. In the literature, these fuzzy inputs and outputs are often represented by IV-T1FSs. When considering a parametric model, the parameter estimation problem is traditionally approached by the minimization of an objective function (with or without a penalty term and sometimes under constraints). In this framework, the vector of parameters that minimizes this objective function is considered as the optimal one. Therefore, parameter estimation plays an important role in regression problems. In fuzzy regression framework, two main methods of parametric regression are considered: the possibilistic methods and least squares (LS) approaches. In contrast with these approaches, the method proposed here is not based on the minimization of an objective function. Instead of determining a single optimal vector of parameters, the suggested approach aims at determining the set of all feasible vectors of parameters. This set of solutions is in concordance with the model structure, with the used fuzzy input/output data and their inherent uncertainties.

Let us consider the nonlinear regressive model given by:

$$y = f(a_0, a_1, x) = a_0 \times \exp(-a_1 \times x^2)$$

where $a = (a_0, a_1)$ is the vector of parameters, and x and y are the model input and model output, respectively. Let us assume that this model is uncertain with fuzzy inputs and fuzzy outputs that are represented by IV-T1FSs. A process is realized where at each a fuzzy input X_i , a fuzzy output Y_i is collected. For instance, this process is repeated 5 times. The data are shown in Table 1 where the inputs are triangular IV-T1FSs and the outputs are trapezoidal IV-T1FSs.

The objective is to determine the set of all feasible parameters. In this estimation problem, at each level α , two CSs are defined. The CS $\mathbb{A}^{\text{sup}}(\alpha)$ represents the plausible parameters set where there is at least one x_i ($\forall i \in \{1, \dots, 5\}$) in $[X_i(\alpha)]$, which guarantees that $f(a_0, a_1, x_i) \in [Y_i(\alpha)]$. The CS $\mathbb{A}^{\text{inf}}(\alpha)$ refers to the set of parameters that certifies that, for all x_i

($\forall i \in \{1, \dots, 5\}$) in $[X_i(\alpha)]$, $f(a_0, a_1, x_i) \in [Y_i(\alpha)]$ is ensured. The results for $\alpha = 0$ and $\alpha = 1$ are illustrated in Figs. 27-28.

Table 1: inputs and outputs IV-TIFSs

i	fuzzy input: $[X_i(\alpha)]$	fuzzy output: $[Y_i(\alpha)]$
1	$[0.01+0.04\alpha, 0.07-0.02\alpha]$	$[4+\alpha, 10-2\alpha]$
2	$[0.09+0.02\alpha, 0.13-0.02\alpha]$	$[2+2\alpha, 8-\alpha]$
3	$[0.27+0.03\alpha, 0.34-0.04\alpha]$	$[\alpha, 5-2\alpha]$
4	$[0.15+0.02\alpha, 0.2-0.03\alpha]$	$[2+\alpha, 6-\alpha]$
5	$[0.22+0.02\alpha, 0.28-0.04\alpha]$	$[1+\alpha, 4-\alpha]$

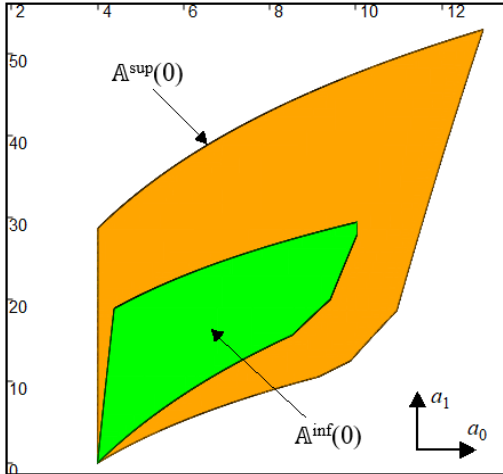


Fig. 27: The TS $[[A(0)]] = [[A^{\text{inf}}(0), A^{\text{sup}}(0)]]$

The sets $A^{\text{inf}}(\alpha)$ and $A^{\text{sup}}(\alpha)$ are computed by the expressions:

$$\begin{aligned}
 A^{\text{inf}}(\alpha) &= \{ \mathbf{a} = (a_0, a_1) \mid \forall i \in \{1, \dots, 5\}, \\
 &\quad \forall x_i \in [X_i(\alpha)], f(a_0, a_1, x_i) \in [Y_i(\alpha)] \} \\
 &= \bigcap_{i \in \{1, \dots, 5\}} \{ \mathbf{a} = (a_0, a_1) \mid \forall x_i \in [X_i(\alpha)], f(a_0, a_1, x_i) \in [Y_i(\alpha)] \} \\
 &= \bigcap_{i \in \{1, \dots, 5\}} \{ \mathbf{a} = (a_0, a_1) \mid \exists x_i \in [Y_i(\alpha)], f(a_0, a_1, x_i) \notin [Y_i(\alpha)] \} \\
 A^{\text{sup}}(\alpha) &= \{ \mathbf{a} = (a_0, a_1) \mid \forall i \in \{1, \dots, 5\}, \\
 &\quad \exists x_i \in [X_i(\alpha)], f(a_0, a_1, x_i) \in [X_i(\alpha)] \} \\
 &= \bigcap_{i \in \{1, \dots, 5\}} \{ \mathbf{a} = (a_0, a_1) \mid \exists x_i \in [X_i(\alpha)], f(a_0, a_1, x_i) \in [Y_i(\lambda)] \}
 \end{aligned}$$

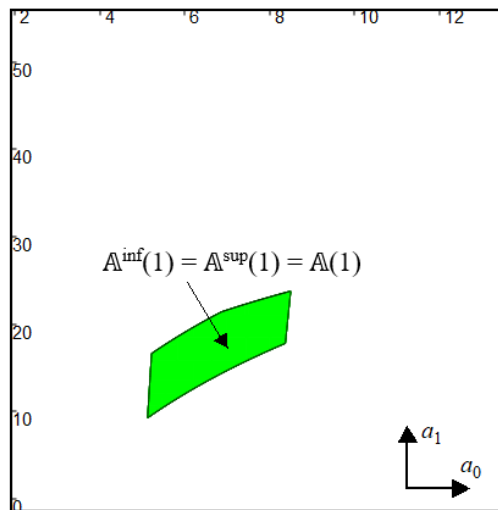


Fig. 28: The CS $[[A(1)]] = A(1)$

At each level α , the set of parameters is given by the TS $[[A(\alpha)]] = [A^{\text{inf}}(\alpha), A^{\text{sup}}(\alpha)]$. The uncertainty for the parameters is represented by the penumbra $A^{\text{sup}}(\alpha) \setminus A^{\text{inf}}(\alpha)$, which represents the set of plausible but not certain parameters. It can

be stated that, when the input is crisp (at $\alpha = 1$ because the inputs are triangular IV-TIFSs), the set of parameters is a CS—i.e., $A^{\text{inf}}(1) = A^{\text{sup}}(1) = A(1)$ (see Fig. 28).

By integrating the dimension $\alpha \in [0, 1]$, the stacking of TSs leads to the TFS which is illustrated in Fig. 29 when a sampling step size of 0.2 on α is used.

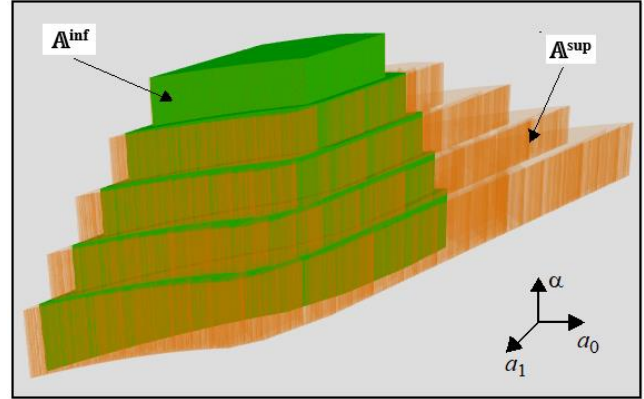


Fig. 29: The TFS $[[A]] = [[A^{\text{inf}}, A^{\text{sup}}]]$

C. Characterisation of a zone explored by a robot

This section illustrates an application of our approach on a real experiment made by the Daurade underwater robot (see Fig. 30). The characteristics of Daurade are as follows:

weight: 1010kg, length: 5m, speed: up to 8 knots (4.11m/s), max depth: 300m, autonomy: 10h at 4 knots, 2h at 8 knots, sonar coverage range: 150m. This robot has been built by ECA robotics and used by DGA Tn (Direction Générale de l'Armement - Techniques Navales – French Army) and SHOM (Service Hydrographique et Océanographique de la Marine) for performing REA (Rapid Environment Assessment) studies. REA is intended to survey the environmental conditions of a particular location in order to identify any existing or potential dangers. In the counter-mine warfare context, attention is focused on mapping the seafloor with acoustic sensors. Daurade is equipped with a Side Scan Sonar (Klein 5500) used to detect potential mines. With this sonar, data are recorded on a line perpendicular to the path of the sensor, and images are formed by putting side by side these lines. The portside lateral sonar antenna corresponds to the one-meter black segment at the bottom left of the robot.

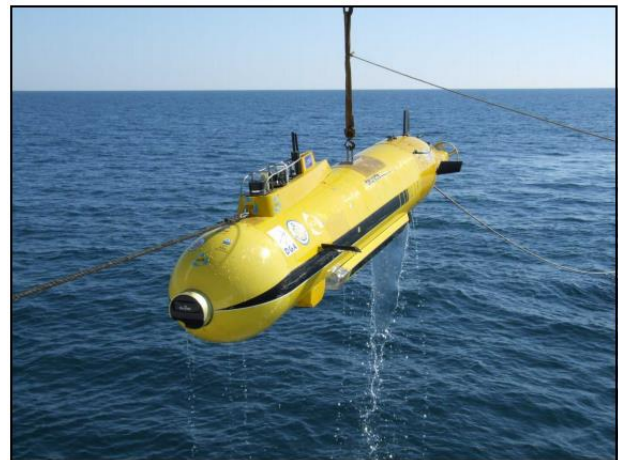


Fig. 30: The Daurade underwater robot used for the experiment.

The experience is a mission of 46 minutes which has been performed in the Roadstead of Brest (France, Brittany) with Daurade. It realized a classical survey pattern composed of a set of parallel tracks with a depth of about 10 meters. The position of the Daurade underwater robot is given by its 2D horizontal coordinates $\mathbf{x} = (x_1, x_2)$. For the navigation, Daurade relies on an inertial unit (Phins II IXBlue) coupled with a DVL (Doppler Velocity Log) sensing speed, which returns after integration the 2D horizontal position coordinates (x_1, x_2) of the robot with respect to the ground (the depth is fixed at 10 m). Once under the water, no GPS data are available and the estimated position of the robot drifts with the time. A key point of REA missions is to estimate if the area of interest has been totally explored without any hole. The approach proposed in this paper can be used to validate that the area to be explored has indeed been covered.

At the beginning of the mission, the position of the robot is exactly known thanks to the GPS localization. The initial condition $\mathbf{x}(0)$ is assumed equal to $(0,0,10)$. The 2D coordinates $(0,0)$ for (x_1, x_2) are considered as our origin of navigation. The Daurade robot is controlled to follow the ideal 2D desired trajectory depicted in Fig. 31 (solid line). In this case, when the robot dives under the surface, it does not receive electromagnetic waves anymore and the GPS cannot be considered. The Daurade robot can estimate its successive positions. However, due to some disturbances (noise sensors in a hostile environment, impact of ocean currents, ...) the robot is subject to some drifting effects on its speed, its heading, and consequently on its position. For instance, an experiment led to the trajectory shown in Fig 31 (dashed line).

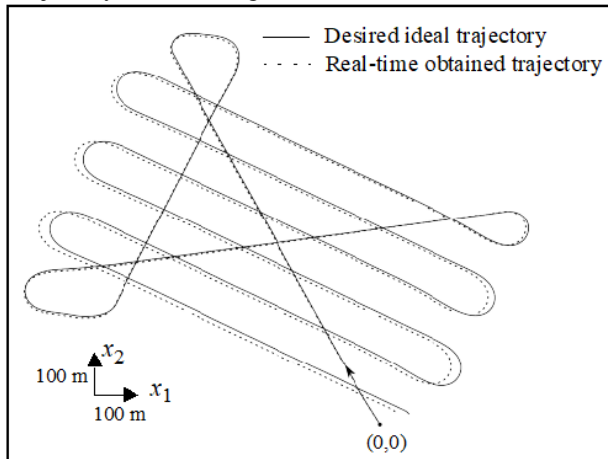


Fig. 31: Desired and obtained real-time trajectories

The uncertainties (the errors) in the robot trajectory are due to the fact that the speed and the heading are uncertain (drift effects). For instance, when a drift on the speed of up to $\pm 0.2\%$ is assumed, the 2D robot trajectory of Fig. 31 (solid line) becomes uncertain and can be represented by a tube \mathcal{X} (see Fig. 32) [36]. This tube represents all possible trajectories of the robot when the speed error is within the interval $[-0.2\%, +0.2\%]$. In this situation, for any considered operating conditions which are compatible with a speed drift of $\pm 0.2\%$, this tube certainly contains the true trajectory. As the robot position is obtained by an integration of its speed, it is normal for the uncertainty of the trajectory to increase over time.

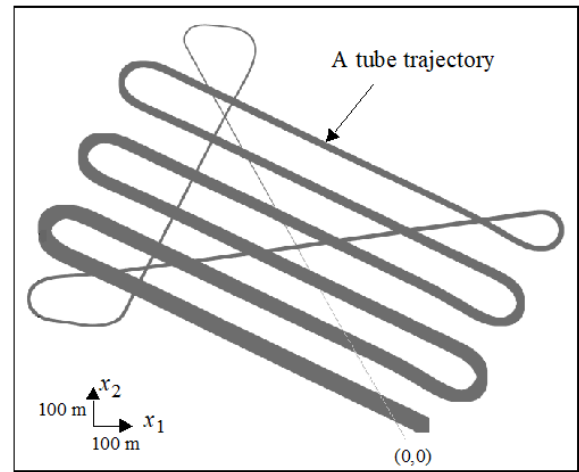


Fig. 32: A tube trajectory: an uncertain trajectory

• **Definition of the explored zone**

The problem to be considered here is the characterization of the explored zone \mathbb{Z} when the robot evolves in a tube (an uncertain trajectory). In its navigation, a scanner on the robot can observe a part of its environment. More precisely, for each time, there exists a subset of the seafloor which is denoted $\mathbb{V}(\mathbf{x}(t)) \subset \mathbb{R}^2$ that is visible by the robot (at each time, the visibility zone can be considered as a disk around the robot position). The set \mathbb{Z} is thus defined by the expression:

$$\mathbb{Z} = \bigcup_{t \geq 0} \mathbb{V}(\mathbf{x}(t))$$

Therefore, due to the uncertain trajectory of the robot, the explored zone \mathbb{Z} cannot be computed exactly and can be represented by a TS, i.e., $\mathbb{Z}^{\text{inf}} \subset \mathbb{Z} \subset \mathbb{Z}^{\text{sup}}$, where:

$$\mathbb{Z}^{\text{inf}} = \bigcap_{\mathbf{x}(\cdot) \in \mathcal{X}(\cdot)} \bigcup_{t \geq 0} \mathbb{V}(\mathbf{x}(t)); \quad \mathbb{Z}^{\text{sup}} = \bigcup_{\mathbf{x}(\cdot) \in \mathcal{X}(\cdot)} \bigcup_{t \geq 0} \mathbb{V}(\mathbf{x}(t))$$

The CS \mathbb{Z}^{inf} is called the certainly explored zone. It corresponds to the set of all points of the environment that have certainly been seen by the robot (by the sonar) taking into account that its trajectory is feasible. The complementary CS of \mathbb{Z}^{sup} is called the certainly unexplored zone. The penumbra is the CS $\mathbb{Z}^? = \mathbb{Z}^{\text{sup}} \setminus \mathbb{Z}^{\text{inf}}$. It corresponds to the set of all points of the environment that have been seen by the sonar for some feasible trajectories and not seen by the sonar for some other feasible trajectories.

• **Estimation of the TFS explored zone**

The speed and heading drifts depend on many factors, more particularly on the noise of the sensors in a hostile environment, on sea currents and weather conditions. For confidentiality purposes, the heading drift is not presented here and only the speed drift is detailed. To determine uncertainties related to the speed drift, experiences are made in adverse (pessimistic) and favorable (optimistic) operating conditions. In the pessimistic case, a drift of up to $\pm 1\%$ has been observed on the robot speed. In the optimistic case, the speed drift is limited to $\pm 0.2\%$. Therefore, in the pessimistic case, the data is the most imprecise but with the highest degree of certainty, i.e. a degree of confidence $1-\alpha = 1$ (level $\alpha = 0$). However, the optimistic situation implies that the data is the most precise but

with the highest degree of uncertainty. This situation refers to $\alpha = 1$ (0-confidence level). For simplicity of implementation, the evolution between the levels 0 and 1 is assumed to be linear. This assumption is only an approximation and supplementary information can be used to improve this linear representation. Therefore, the uncertainty related to the speed drift (the error speed) is characterized by an IV-TIFS given by its α -cuts representation:

$$[E_{speed}(\alpha)] = [-0.01 + 0.008 \cdot \alpha, 0.01 - 0.008 \cdot \alpha]$$

For each level α , $[E_{speed}(\alpha)]$ is a conventional interval and the 2D robot trajectory is uncertain and is represented by a tube. In this case, for each level α , the explored zone is a TS $[[Z(\alpha)]] = [Z^{inf}(\alpha), Z^{sup}(\alpha)]$. For instance, the resulting explored zone and the robot trajectory are depicted in Fig. 33 and Fig. 34 for $\alpha = 1$ and $\alpha = 0$, respectively.

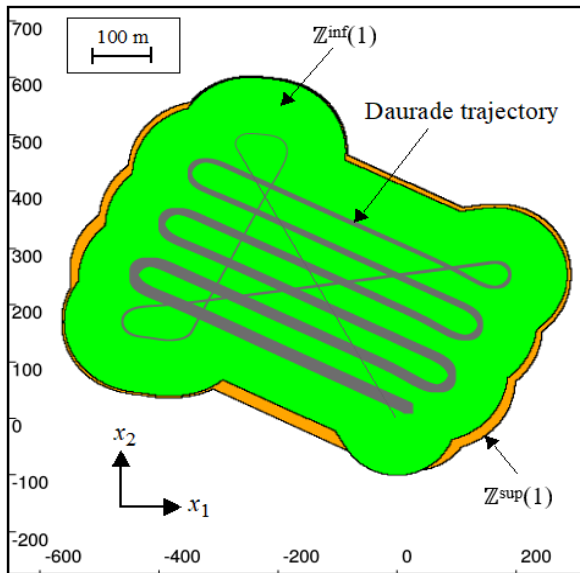


Fig. 33: The TS $[[Z(1)]] = [Z^{inf}(1), Z^{sup}(1)]$ and the uncertain trajectory

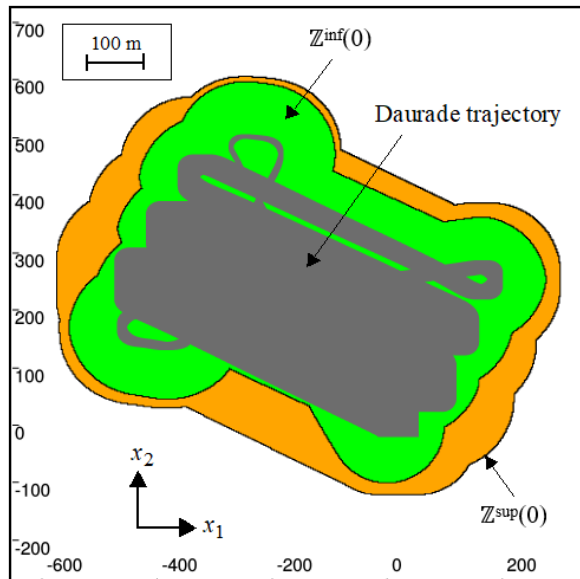


Fig. 34: The TS $[[Z(0)]] = [Z^{inf}(0), Z^{sup}(0)]$ and the uncertain trajectory

For each α -cut level and whatever the position of the robot in its tube (its uncertain trajectory), all the points in $Z^{inf}(\alpha)$ are seen by the sonar with certainty. Moreover, certainly the points

which are outside $Z^{sup}(\alpha)$ are not seen the sonar. The points in the penumbra $Z^{sup}(\alpha) \setminus Z^{inf}(\alpha)$ are possibly (but not certain) seen by the sonar. By integrating the dimension $\alpha \in [0, 1]$, the stacking of TSs leads to the TFS which is illustrated in Fig. 35 when a sampling step size of 0.2 on α is used.

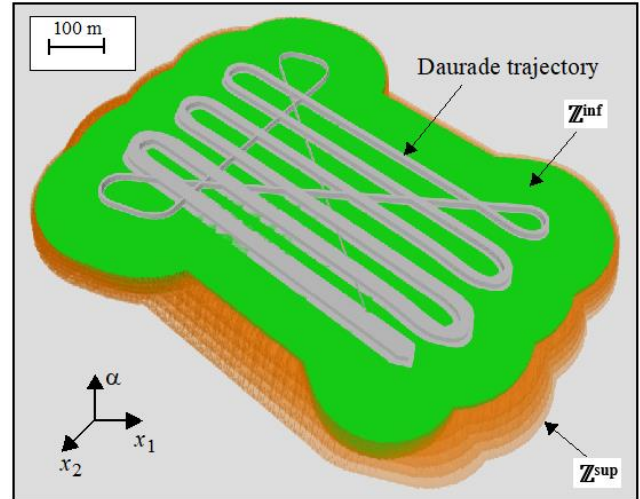


Fig. 35: The TFS $[[Z]] = [Z^{inf}, Z^{sup}]$ and the uncertain trajectory

VI. REMARKS AND DISCUSSIONS

- A TFS is built for modeling uncertainty in TIFSs where the manipulated information is uncertain and cannot be revealed by only a unique TIFS. Moreover, in the absence of uncertainty, a TFS $[[X]]$ becomes a TIFS X , i.e., $X^{inf} = X^{sup} = X$. For instance, and as illustrated in the first application in section V, when a TIFS view based on IA is applied, only the plausible TIFS is obtained. Therefore, if a certain TIFS solution is needed (for safety or reliability purposes for example), the TIFS view (based on α -cuts principle) is not sufficient. The same remark can be made for the second application in estimating the parameters of a nonlinear regressive model. More generally, the concept of TFSs is suitable for uncertain fuzzy problems through a set-method approach where the exhibition of certain and plausible situations is needed for safety and reliability purposes.

- The concept of TSs is relatively close to the concept of rough sets (RSs) [37][38] (see also [39] a survey on rough set theory). A rough set provides a representation of a given CS using lower and upper approximations when the available information is not sufficient for determining the exact boundaries of this CS. Therefore, a RS X is approximately represented by two CSs, called lower and upper approximations. In this case, in a discrete context, given the lower approximation (objects fully classified as X) and upper approximation (objects possibly classified as X), the boundary region of X can be constructed. So, the uncertain set X is characterized by the boundary region interpreted as a RS (objects which can be classified neither as X nor as to its complement). It consists of objects that are not inside or outside X . The boundary region is the difference between the upper and lower approximations. If the boundary region is empty, X is considered as CS.

In rough set theory, information is often presented as a data-table, columns of which are labeled by attributes, rows by objects of interest and entries of the table are attribute values. In its conception, if the concept of TS can be close to the concept of RS, their finality and their field of application differ. In rough set theory, the approximation of a CS by a pair of lower and upper CSs is elaborated using an equivalence relation defined by a set of attributes. In the proposed approach, a TS is interpreted as an uncertain set and is represented by an interval of CSs where its bounds are regarded as certain and plausible. Furthermore, unlike RSs, TSs computations are implemented using interval-based arithmetic and solvers.

- As explained in the paper introduction, TFSs use a set-method approach based on α -cuts principle for their representation. Therefore, in this paper, all the figures illustrating TFSs are not membership functions but a family of stacked and nested TSs according to the dimension α . The equivalence between the functional view (based on membership functions) and the set-method representation (based on CSs stacking) can be obtained for particular shapes of T1FSs and T2FSs with single variables such as IV-T1FSs and IV-T2FSs. However, the formalization and the generalization of this equivalence to higher-order T1FSs and T2FSs with multiple variables remain very difficult or even impossible. This is why the set-method view has been favored over a functional approach.

- Compared to T2FSs, IV-T2FSs have the advantage of simplicity in computer implementation and can take benefit from the arsenal of IA for their computations. In this paper, an IV-T2FSs is regarded as a family of nested TIs. Each TI is represented by lower and upper conventional intervals. This representation, although useful, does not allow a direct extension of standard IA (SIA) [40] to IV-T2FSs. To remedy this situation and to permit a direct extension of SIA methods to IV-T2FSs another but equivalent representation has been proposed in [41]. This representation is not based on lower and upper intervals bounds but on the left and right bounds.

- The construction of a TFS requires that the TSs obtained by α -cuts are nested. Thus, obtaining a TFS imposes the monotonicity (consistency) constraint. However, in some practical situations where the constraint of monotonicity can be relaxed, it is possible to obtain TSs which are not necessarily nested. In this case, the lower and upper bounds are no longer T1FSs but gradual sets [42]. In this framework, the TFS becomes a TGS and it cannot be seen as a T2FS.

VII. CONCLUSION

In this paper, the concept of TFSs is proposed. According to the α -cut principle, a TFS is regarded as a family of TSs. In this context, a TFS is an interval of T1FSs, delimited by two lower and upper bounds. The TFS concept can give an alternative representation of uncertain T1FSs using TSs. Therefore, knowing that a TFS is represented by two T1FSs (lower and upper T1FSs) under the constraint that the lower

T1FS \subset the upper T1FS, it involves that a TFS could be regarded as a T2FS. According to the TFS idea, uncertain fuzzy computations can be made on intervals of T1FSs according to interval-based tools and solvers. The essence and the potential applicability of TFSs have been validated through academic examples and a real-world application for modeling the zone explored by an underwater robot. Furthermore, the TFSs can be applied to more applications in many domains such as fuzzy modeling, linear and nonlinear uncertain fuzzy control, among others. Future papers will be dedicated to these interesting research directions, especially in fuzzy control applications.

APPENDIX

A. Projection principle

In this appendix the methodology used for computing \mathbb{Z}^{inf} and \mathbb{Z}^{sup} of application example 1 is shown. Let us consider the 4-dimensional CS:

$$\mathbb{A} = \{(\mathbf{x}, \mathbf{p}) \in \mathfrak{R}^4 \mid \|\mathbf{x} - \mathbf{p}\| \leq 35\}$$

and the CS given by:

$$\mathbb{B} = \mathbb{A} \cap (\mathfrak{R}^2 \times [\mathbf{p}]) = \{(\mathbf{x}, \mathbf{p}) \in \mathfrak{R}^4 \mid \mathbf{p} \in [\mathbf{p}] \text{ and } \|\mathbf{x} - \mathbf{p}\| \leq 35\}$$

The CS \mathbb{Z}^{sup} corresponds to the projection of \mathbb{B} on the \mathbf{x} -space.

Therefore, \mathbb{Z}^{sup} is computed as follows:

$$\mathbb{Z}^{\text{sup}} = \text{proj}_{\mathbf{x}} \mathbb{B} = \text{proj}_{\mathbf{x}} \mathbb{A} \cap (\mathfrak{R}^2 \times [\mathbf{p}])$$

Using the same principle, \mathbb{Z}^{C} is given by:

$$\mathbb{Z}^{\text{inf}} = \overline{\text{proj}_{\mathbf{x}} \mathbb{A} \cap (\mathfrak{R}^2 \times [\mathbf{p}])}$$

The projection operation is implemented using the method proposed in [32].

B. Implementation of example 1 in section II.C using PyIbex

```

from pyibex import *
from vibes import vibes
f = Function("x1", "x2", "m1", "m2", "(x1-m1)^2+(x2-m2)^2")
Zd_out = SepFwdBwd(f, [-oo, 1225])
# the plausible set
Z_out = SepProj(Zd_out, IntervalVector([-4, 8], [2, 18]))
Zd_in = ~SepFwdBwd(f, [-oo, 1225])
# the certain set
Z_in = ~SepProj(Zd_in, IntervalVector([-4, 8], [2, 18]))
vibes.beginDrawing()
vibes.newFigure()
vibes.axisLimits(-50, 45, -35, 60, figure="")
vibes.axisLabels("", "", figure="")
# plot the certain set
params={'color_in':'green[green]',
        'color_out':transparent[transparent]',
        'color_maybe':'black[black]', 'use_patch': False}
pySIVIA([-50, 45], [-35, 60], Z_in, 0.1, draw_boxes=True,
        save_result=False, **params)
#plot the penumbra
params={'color_in':'orange[orange]',
        'color_out':transparent[transparent]',
        'color_maybe':'black[black]', 'use_patch': False}
    
```

```
pySIVIA([-50,45],[-35, 60]), Z_out&~Z_in, 0.1,  
    draw_boxes=True, save_result = False, **params)  
#plot the plausible set  
params = {'color_in': 'transparent[transparent]',  
    'color_out':'transparent[transparent]',  
    'color_maybe':'black[black]', 'use_patch': False}  
pySIVIA([-50, 45],[-35, 60]),Z_out,0.1,draw_boxes=True,  
    save_result=False, **params)  
#plot the box  
vibes.drawBox(-4,8, 2,18,'black[black]')
```

REFERENCES

- [1] Zadeh L. A., The concept of a linguistic variable and its application to approximate reasoning – 1, *Inform. Sci.* vol. 8, pp. 199–249, 1975.
- [2] Mendel J.M. and Bob John R.I., Type-2 Fuzzy Sets Made Simple, *IEEE Transactions on Fuzzy Systems*, Vol. 10(2), pp. 117-127, 2002.
- [3] Mendel J.M., *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*. Upper Saddle River, NJ: Prentice-Hall, 2001.
- [4] Mendel J.M., John R.I. and Liu F., Interval type-2 fuzzy logic systems made simple, *IEEE Trans. Fuzzy Syst.* Vol. 14 (6), pp. 808-821, 2006.
- [5] Mendel J.M., Wu H., Type-2 Fuzzistics for Symmetric Interval Type-2 Fuzzy Sets: Part 1, Forward Problems, *IEEE Trans. On Fuzzy syst.*, Vol. 14 (6), pp. 781-792, 2006.
- [6] Mendel J. M., Rajati M. R., and Sasser P., On clarifying some definitions and notations used for type-2 Fuzzy sets as well as some recommended notational changes, *Info. Sci.*, Vol. 340-341, pp. 337-345, 2016.
- [7] Castillo O., Cervantes L., Soria J., Sanchez M. and Castro J-R., Generalized type-2 fuzzy granular approach with applications to aerospace, *Information Sciences*, Vol. 354, pp. 165–177, 2016.
- [8] Castillo O., Amador-Angulo L., Castro J-R. and Garcia-Valdez M., A comparative study of type-1 fuzzy logic systems, interval type-2 fuzzy logic systems and generalized type-2 fuzzy logic systems in control problems, *Information Sciences*, Vol. 354, pp. 257-274, 2016.
- [9] Sanchez M-A., Castillo O. and Castro J-R., a Generalized Type-2 Fuzzy Systems for controlling a mobile robot and a performance comparison with Interval Type-2 and Type-1 Fuzzy Systems, *Expert Systems with Applications*, Vol. 42, pp. 5904–5914, 2015.
- [10] Mendel J.M., Type-2 fuzzy sets and systems: a retrospective, *Informatik-Spektrum*, Vol. 38, pp. 523–532, 2015.
- [11] Chen, T. Y., An ELECTRE-based outranking method for multiple criteria group decision making using interval type-2 fuzzy sets, *Information Sciences*, Vol. 263 , pp. 1–21, 2014.
- [12] Wu D. and Mendel J. M., Aggregation Using the Linguistic Weighted Average and Interval Type-2 Fuzzy Sets, *IEEE Transactions on Fuzzy Systems*, Vol. 15 (6), pp. 1145-1161, 2007.
- [13] Zhou S.M., John R. I., Chiclana F., and Garibaldi J. M., On aggregating uncertain information by type-2 OWA operators for soft decision making,” *Inter. J. of Intelligent Syst.*, Vol. 25 (6), pp. 540–558, 2010.
- [14] Boukezzoula R., and Coquin D., A decision-making computational methodology for a class of type-2 fuzzy intervals: An interval-based approach, *Information Sciences*, Vol. 510, pp. 256-282, 2020.
- [15] Golsefid, S. M. M., Zarandi, M. H. F., and Turksen, I. B., Multi-central general type-2 fuzzy clustering approach for pattern recognition, *Information Sciences*, Vol. 328, pp. 172–188, 2016.
- [16] Melin, P., Gonzalez, C. I., Castro, J. R., Mendoza, O., and Castillo, O., Edge-detection method for image processing based on generalized type-2 fuzzy logic, *IEEE Trans. on Fuzzy Systems*, Vol. 22 (6), pp. 1515–1525, 2014.
- [17] Ontiveros-Robles E., Melin P. and Castillo O., Comparative analysis of Noise Robustness of Type 2 Fuzzy Logic Controllers, *Kybernetika*, Vol. 54 (1), pp. 175–201, 2018.
- [18] Lam HK., Li H., Deters C., Secco EL., Wurdemann HA., Althoefer K., Control design for interval type-2 fuzzy systems under imperfect premise matching, *IEEE Transactions on Industrial Electronics*, Vol. 61(2), pp. 956-968, 2013.
- [19] Li H., Wang J., Lam HK., Zhou Q., Du H., Adaptive sliding mode control for interval type-2 fuzzy systems, *IEEE Transactions on Systems, Man, and Cyb.: Systems*, Vol. 46(12), pp. 1654-1663, 2016.
- [20] Xiao B., HK Lam HK., Li H., Stabilization of interval type-2 polynomial-fuzzy-model-based control systems, *IEEE Transactions on Fuzzy Systems*, Vol. 25(1), pp. 205-217, 2016.
- [21] Klir, G.J. and Yuan, B., *Fuzzy Sets and Fuzzy Logic, Theory and Applications*, Prentice Hall, New Jersey, 1995.
- [22] Mizumoto M. and Tanaka K., Some properties of fuzzy numbers, in: *Advances in Fuzzy Set Theory and Applications*, Gupta, M.M., Ragade R.K. and Yager R.R. Eds., New York, pp. 153-164, 1979.
- [23] Nguyen, H.T., A note on the extension principle for fuzzy sets, *Journal of Math. Analysis and Applications*, Vol. 64, pp. 369-380, 1978.
- [24] Kaufmann A. and Gupta M.M., *Introduction to fuzzy arithmetic: Theory and Applications*, V. Nostrand Reinhold Comp. Inc. New York, 1991.
- [25] Ralescu D.A., A generalization of the representation theorem, *Fuzzy Sets and Systems*, Vol. 51, pp. 309-311, 1992.
- [26] Uehara K. and Fujise M., Fuzzy inference based on families of α -level sets, *IEEE Transactions on Fuzzy Systems*, Vol. 1, pp. 111-124, 1993.
- [27] Dymova L., Sevastjanov P., and Tikhonenko A., An interval type-2 fuzzy extension of the TOPSIS method using alpha-cuts, *Knowledge-Based Systems*, Vol. 83, pp. 116–127, 2015.
- [28] Hamrawi H., S. Coupland, and R. John, Type-2 Fuzzy Alpha-Cuts, *IEEE Transactions on Fuzzy systems*, Vol. 25 (3), pp. 682-692, 2017.
- [29] Liu F., An efficient centroid type reduction strategy for general type-2 fuzzy logic system, *Inf. Sci.*, Vol. 178, pp. 2224-2236, 2008.
- [30] Desrochers B., and Jaulin L., Thick set inversion, *Artificial Intelligence*, Vol. 249, pp. 1-18, 2017.
- [31] Desrochers B., *Simultaneous Localization and Mapping in Unstructured Environments, A Set-Membership Approach*, PhD thesis, ENSTA Bretagne, France, 2018.
- [32] Jaulin L., Braems I. and Walter E., Interval methods for nonlinear identification and robust control, In *Proc. of the 41st IEEE Conference on Decision and Control*, Las Vegas, Vol. 4, pp. 4676-4681, 2002.
- [33] Chabert G. and Jaulin L., Contractor programming, *Artificial Intelligence*, Vol. 173, pp. 1079-1100, 2009.
- [34] Negoita, C.V. and Ralescu, D.A., *Applications of Fuzzy Sets to Systems Analysis*, Wiley, New York, 1975.
- [35] Lodwick W.A. and Dubois D., Interval linear systems as a necessary step in fuzzy linear systems, *Fuzzy Sets & Syst.*, Vol. 281, pp. 227-251, 2015.
- [36] Rohou S., Jaulin L., Mihaylova M., Le Bars F., Veres S., Guaranteed Computation of Robots Trajectories, *Robotics and Autonomous Systems*, Vol. 93, pp.76–84, 2017.
- [37] Pawlak Z., Rough sets, *Int. J. Comput. Inf. Sci.*, vol. 11 (5), pp. 341-356, 1982.
- [38] Pawlak Z., *Rough Sets: Theoretical Aspects of Reasoning About Data*. Kluwer Academic Publishers, Norwell, MA, USA.
- [39] Zhang Q., Xie Q. and Wang G., A survey on rough set theory and its applications, *CAAI Transactions on Intelligence Technology*, Vol. 1, N° 4, pp. 323-333, 2016.
- [40] Moore RE (1962) *Interval Arithmetic and Automatic Error Analysis in Digital Computing*. PhD Thesis, Department of Computer Science, Stanford University.
- [41] Boukezzoula R., Jaulin L. and Foulloy L., Thick gradual intervals: An alternative interpretation of type-2 fuzzy intervals and its potential use in type-2 fuzzy computations, *Engineering Applications of Artificial Intelligence*, vol. 85, pp. 691-712, 2019.
- [42] Dubois D. and Prade H., Gradual elements in a fuzzy set, *Soft Computing*, No. 12, pp. 165-175, 2008.